

Speed faults in computation by chemical reaction networks*

Ho-Lin Chen[†] Rachel Cummings[‡] David Doty[§] David Soloveichik[¶]

Abstract

Chemical reaction networks (CRNs) formally model chemistry in a well-mixed solution. Assuming a fixed molecular population size and bimolecular reactions, CRNs are formally equivalent to population protocols, a model of distributed computing introduced by Angluin, Aspnes, Diamadi, Fischer, and Peralta (PODC 2004). The challenge of fast computation by CRNs (or population protocols) is to ensure that there is never a bottleneck “slow” reaction that requires two molecules (agent states) to react (communicate), both of which are present in low ($O(1)$) counts. It is known that CRNs can be fast in expectation by avoiding slow reactions with high probability. However, states may be reachable (with low probability) from which the correct answer may only be computed by executing a slow reaction. We deem such an event a *speed fault*. We show that the problems decidable by CRNs guaranteed to avoid speed faults are precisely the *detection problems*: Boolean combinations of questions of the form “is a certain species present or not?”. This implies, for instance, that no speed fault free CRN could decide whether there are at least two molecules of a certain species, although a CRN could decide this in “fast” expected time — i.e. speed fault free CRNs “can’t count.”

1 Introduction

Understanding the principles of molecular computation is essential to making sense of information processing in biological cellular regulatory networks. Further, in engineering of life-like devices (e.g. “wet robots” that can patrol the blood for cancer cells) we are rapidly approaching the point where we are limited by conceptual understanding: How molecular networks can be programmed to process information and carry out computation subject to the natural constraints of aqueous chemistry is still not well-understood.

A foundational model of chemistry commonly used in natural sciences is that of chemical reaction networks (CRNs), i.e., (finite) sets of chemical reactions such as $A + B \rightarrow A + C$. Subject to discrete semantics (integer number of molecules) the model corresponds to a continuous time, discrete state, Markov process [12]. A state of the system is a vector of non-negative integers specifying the molecular counts of the species (e.g., A, B, C), a reaction can occur only when all its reactants are present, and transitions between states correspond to reactions (i.e., when the above reaction occurs the count of B is decreased by 1 and the count of C increased by 1). The transition rate is proportional to the product of the counts of the reactants. CRNs are widely used to

*The third, and fourth authors were supported by the Molecular Programming Project under NSF grants 0832824 and 1317694, the first author was supported by NSC grant number 101-2221-E-002-122-MY3, the second author was supported by NSF grants CCF-1049899 and CCF-1217770, the third author was supported by a Computing Innovation Fellowship under NSF grant 1019343, NSF grants CCF-1219274 and CCF-1162589, and the fourth author was supported by NIGMS Systems Biology Center grant P50 GM081879.

[†]National Taiwan University, Taipei, Taiwan, holinc@gmail.com

[‡]California Institute of Technology, Pasadena, CA, USA, rachelc@u.northwestern.edu.

[§]California Institute of Technology, Pasadena, CA, USA, ddoty@caltech.edu

[¶]University of California, San Francisco, San Francisco, CA, USA, david.soloveichik@ucsf.edu

describe natural biochemical systems such as the intricate cellular regulatory networks responsible for the information processing within cells. With recent advances in synthetic biology, CRNs are a promising language for the design of artificial biochemical networks. For example, the physical primitive of nucleic-acid strand displacement cascades provides concrete chemical implementations of arbitrary CRNs [4, 8, 17]. Thus, since in principle any CRN can be built, hypothetical CRNs with interesting behaviors are becoming of more than theoretical interest.

The importance of the CRN model is underscored by the observation that intimately related models repeatedly arise in theoretical computer science under different guises: e.g. vector addition systems [13], Petri nets [15], population protocols [1]. The connection to distributed computing models, in turn, resulted in novel insights regarding natural cellular regulatory networks [5].

Parallelism is a basic attribute of chemistry, and one that is of central importance in understanding molecular information processing. This kind of parallelism is both a blessing and a curse: it can be used to speed up computation, but we must be careful to avoid “race conditions” (reactions happening in an unintended order) which may lead to error.

Consider a very basic task: a chemical system (e.g. cell) responding to molecular signals present in very small quantities. More specifically, say the computation is to produce at least one molecule of Y if and only if there is at least one molecule of species A_1 and at least one molecule of species A_2 . Consider the strategy shown in Fig 1(b). Intuitively, this corresponds to having receptors F that in order to activate need to bind both A_1 and A_2 . By having n receptors F we can increase the rate of the first reaction, but if there is only one molecule of A_1 , there will be at most one molecule of F' and thus the second reaction occurs at a rate independent of the amount of receptor. Thus this scheme is “not parallelizable”.¹

A better strategy is to amplify the signal before taking the conjunction: e.g. Fig 1(c). Here the receptors release A_1 back upon interacting with it, and a single A_1 can interact with many receptors (converting them from F to F'). Intuitively, the more receptors F we have, the faster we’ll get a large number of F' ’s, and the faster the Y will get produced via the second reaction. More specifically, observe that starting with $n > 0$ molecules of F , and a molecule of A_1 and A_2 each, the reachable states without Y are: for $0 \leq m \leq n$, $((n - m) F, m F', 1 A_1, 1 A_2)$. From any reachable state without Y , we can reach Y through

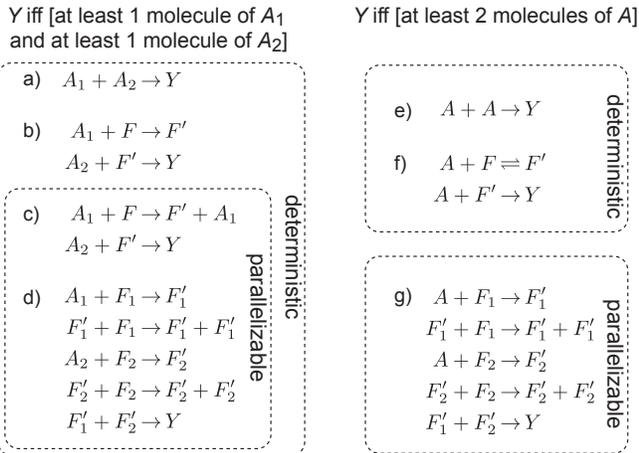


Figure 1: Two molecular computation tasks: predicates “Is there at least 1 molecule of A_1 and at least one molecule of A_2 ?” (left), and “Are there at least 2 molecules of A ?” (right). CRNs (a)-(d) compute the first predicate (left), and CRNs (e)-(g) compute the second (right). Parameter n is the initial amount of F , or F_1 and F_2 species which help in the computation. Informally the parallelizable CRNs are those that produce the output faster with increasing n . Deterministic CRNs are those that compute correctly no matter what order the reactions happen to occur in. Other strategies (not shown) involve producing Y but consuming it if the predicate is not satisfied.

¹Bimolecular reaction rates scale inversely with the total volume, and it is impossible to fit arbitrarily many molecules in a fixed volume. While for large enough molecular counts we will run into this finite density constraint, we study the scaling of speed with molecular count before that point is reached. An alternate perspective is that our task is to compute as quickly as possible in volume sufficient to allow molecules of F to fill the volume with constant density [16].

a sequence of reaction executions where one of the reactants is present in at least $\lfloor n^{1/2} \rfloor$ count,² and under stochastic chemical kinetics, the expected time to traverse this path is $O(1/n^{1/2})$ — decreasing with n .³ Scheme Fig 1(d) is even faster: it can be shown that from any reachable state, the expected time to produce Y scales as $O(\log n/n)$.

Now consider a slightly different computational task: produce at least one molecule of Y if and only if there are at least 2 molecules of species A . The natural analog of Fig 1(b) fails to be deterministic: the reactions $A + F \rightarrow F'$, $A + F' \rightarrow Y$ suffer from a “race condition” where Y is never produced if both molecules of A happen to react with F . This can be fixed by having the receptor F bind A reversibly⁴ as in Fig. 1(f). However, this scheme is not parallelizeable for the same reason as (b).

The natural analog of the parallelizeable reaction scheme Fig 1(c) will not solve this task correctly at all: With reactions $A + F \rightarrow F' + A$, $A + F' \rightarrow Y$, even a single molecule of A will always lead to a Y .

Also problematic is the scheme shown in Fig 1(g) based on (d). While it is parallelizeable, it also suffers from a race condition that can result in an error. If the two molecules of A happen to react with different receptor types (F_1 and F_2) then Y will be produced. However, if both A ’s react with the same receptor type, Y will never be produced.

Informally, our main result is that no CRN is deterministic and parallelizeable at the same time for the “2 A problem” (or any computation that involves counting, rather than simply detecting the presence or absence of input species). Thus deterministic and parallelizeable must be disjoint in Fig. 1(right). Unlike the examples above, we allow a broader range of schemes that could produce and consume Y repeatedly but eventually converge on the presence or absence of Y as the output. In order to define “parallelizeable” formally, we introduce the notion of a “speed fault”. A speed fault occurs if a state is reached such that to stabilize to the correct output from that state requires using a bimolecular reaction with both reactants bounded independently of n . Thus “deterministically parallelizeable” corresponds to speed fault free. Our main result is that the problems decidable by speed fault free CRNs are precisely the *detection problems*: Boolean combinations of questions of the form “is a certain species present or not?”. Thus speed fault free CRNs “can’t count.”

The current work stems from the desire to understand fast deterministic computation in CRNs and population protocols. While sophisticated chemical algorithms and protocols have been developed to compute a large class of functions quickly and without error (see next section), most constructions are not deterministically fast in the same strong sense as they are deterministic. Indeed, deterministic computation is a worst case notion that intuitively ensures correctness no matter what unlucky sequence of reactions occurs. However, fast computation is defined with respect to large probability reaction sequences. Our definition captures the natural worst case notion of speed.⁵

²If $m < \lfloor n^{1/2} \rfloor$, execute the first reaction $\lfloor n^{1/2} \rfloor - m$ times (resulting in $\lfloor n^{1/2} \rfloor$ molecules of F'), and then execute the second reaction. If $m \geq \lfloor n^{1/2} \rfloor$, execute the second reaction.

³The rate of a bimolecular reaction is proportional to the product of the counts of the reactants. Thus the expected time from the state with $m < \lfloor n^{1/2} \rfloor$ molecules of F' to reach the state with $\lfloor n^{1/2} \rfloor$ molecules of F' is proportional to $\sum_{i=m}^{\lfloor n^{1/2} \rfloor} 1/(n-i) \leq n^{1/2} \cdot 1/(n-n^{1/2}) = O(1/n^{1/2})$. Finally the rate of the second reaction when there are $\lfloor n^{1/2} \rfloor$ molecules of F' is proportional to $n^{1/2}$ and thus the expected time for it to fire is $O(1/n^{1/2})$ for a total expected time of $O(1/n^{1/2})$. Note that power $n^{1/2}$ was chosen in the analysis to ensure the optimal tradeoff between the rates of individual reaction executions and the total number of reaction executions.

⁴A reversible reaction $A+F \rightleftharpoons F'$ is simply syntactic sugar for two irreversible reactions $A+F \rightarrow F'$ and $F' \rightarrow A+F$.

⁵We observe that in the literature on computation in CRNs and population protocols it is almost never the case that computation is slow because the necessary sequence of reactions is too long – rather, slowdown is dominated by reaction bottlenecks where two low count species must react. Thus in this work we focus on this essential type of delay, captured in our notion of speed faults.

Our positive result shows how any detection problem can be decided by a speed fault free CRN, and further shows that this computation is fast in the standard stochastic chemical kinetics model [12]. The largest part of this paper concerns the negative result that only detection problems can be computed by speed fault free CRNs (Section 4.2). The proof of the negative result consists of finding a worst-case reaction sequence that leads to a speed fault, assuming a non-detection problem is computed.

Absent speed-faults, the $O(1)$ -count species must initiate cascades through intermediary large count species in order to “communicate.” Consider the above “ $2A$ problem.” We can imagine isolating the two copies of A in “separate test tubes” and then use the symmetry between the two A molecules to make the system think that it’s communicating with just one A (and thereby fail to detect the second A). To make this argument precise we develop a pumping technique which formally distinguishes species that can get arbitrarily large with increasing n from species whose counts are bounded by a constant⁶. We show that all large count species that can be encountered along a trajectory can be pumped to be *simultaneously* large. We then show that in the context of large counts of all pumpable species, reaction sequences can be decomposed into separate test tubes (parallel decomposition). A key part of the argument involves showing that the speed fault free CRN cannot detect small changes to pumpable species; for this we develop a new technique for performing surgery on reaction sequences.

2 Previous work and future directions

Much related work in the distributed computing community is phrased in the language of population protocols rather than CRNs (e.g. [2]). While population protocols are equivalent to CRNs with exactly two reactants and two products, and thus a fixed population size, CRNs can naturally describe reactions that consume or produce net molecules. As a result CRNs can potentially explore an unbounded state space, and certain questions that are not natural for population protocols become germane for CRNs (for example: Turing universality). Because our negative result naturally applies to a changing population size, we phrase this paper in the language of CRNs.

CRNs have a surprisingly rich computational structure. If we allow the number of species and reactions to scale with the size of the input (i.e. we view CRNs as a non-uniform model of computation), then $\log s$ species can deterministically simulate space s -bounded Turing machines [6]. (These results are presented in a model called vector addition systems [13], but easily carry over.) Thus CRNs are a very powerful model of non-uniform computation. On the other hand, we ask what functions can be computed by a fixed CRN (i.e. fixed number of species and reactions, with input encoded in the initial molecular counts, which corresponds to a uniform model). In this setting, CRNs are not Turing universal, unless we allow for some probability of error [3, 16]. In attempting Turing universal computation, there will provably always be “race conditions” that lead to error if certain reactions occur in a (maybe unlikely but possible) malicious order. The fact that even such Turing universal computation is possible, and indeed can be made “fast” is surprising since finite CRNs necessarily must represent binary data strings in a unary encoding, since they lack positional information to tell the difference between two molecules of the same species.

Deterministic computation of both predicates and functions has been exactly characterized, and corresponds to semilinear sets and functions [2, 7]. Angluin, Aspnes, and Eisenstat [2] showed that all semilinear predicates can be deterministically computed in expected $O(n \text{ polylog } n)$ “interactions” (molecules bumping into each other). In a volume of fixed size, with n molecules, there are

⁶Note that our pumping lemma is very different from a similarly called “pumping lemma” of ref. [2], which shows that how input can be increased without changing the output (thus pumping “input”)

an expected $\Theta(n^2)$ such interactions per unit time, which yields expected time $O((1/n)\text{polylog } n)$ — decreasing with n . Our results imply that when computing semilinear predicates other than the detection problems, it is always possible to reach a state (speed fault) from which the expected time to finish the computation is $\Omega(1)$ — independent of n . It is easy to reconcile the two results: in the construction of ref. [2], the probability that a speed fault is reached decreases with n , and thus the total expected time decreases with n as well. Our result implies that this is a necessary feature of any such construction, and is not simply due to insufficient cleverness of the researchers to avoid speed faults.

Other work showing the challenges in parallelizing CRNs include the investigation of running multiple copies of networks in parallel [9], and the inability of networks starting with only large count species to delay the production of any species [11].

While in this work we focused on parallelizable predicates, it remains to explore the class of parallelizable functions. For example, if the initial amount of A is the input and the final amount of B is the output, then we can think of the reaction $F + A \rightarrow 2B$ as deterministically computing $f(x) = 2x$. Clearly as the amount of F increases, the computation converges faster. On the other hand, we believe that computing division by 2 should not be possible without speed faults, although that remains to be shown.

Since the occurrence of a speed fault leads to a slow computational bottleneck, speed faults affect the tail bounds on the distribution of the computation time. Indeed, two CRNs may compute with the same fast expected time, but the one susceptible to speed faults will likely have a larger probability of taking significantly longer. It remains to rigorously draw out the connection between tail bounds and speed faults.

3 Preliminaries

3.1 Chemical reaction networks

If Λ is a finite set (in this paper, of chemical species), we write \mathbb{N}^Λ to denote the set of functions $f : \Lambda \rightarrow \mathbb{N}$. Equivalently, we view an element $\mathbf{c} \in \mathbb{N}^\Lambda$ as a vector of $|\Lambda|$ nonnegative integers, with each coordinate “labeled” by an element of Λ . Given $S \in \Lambda$ and $\mathbf{c} \in \mathbb{N}^\Lambda$, we refer to $\mathbf{c}(S)$ as the *count of S in \mathbf{c}* . Let $\|\mathbf{c}\| = \|\mathbf{c}\|_\infty = \max_{S \in \Lambda} \mathbf{c}(S)$. We write $\mathbf{c} \leq \mathbf{c}'$ to denote that $\mathbf{c}(S) \leq \mathbf{c}'(S)$ for all $S \in \Lambda$, and $\mathbf{c} < \mathbf{c}'$ if $\mathbf{c} \leq \mathbf{c}'$ and $\mathbf{c} \neq \mathbf{c}'$. Since we view vectors $\mathbf{c} \in \mathbb{N}^\Lambda$ equivalently as multisets of elements from Λ , if $\mathbf{c} \leq \mathbf{c}'$ we say \mathbf{c} is a *subset* of \mathbf{c}' . Given $\mathbf{c}, \mathbf{c}' \in \mathbb{N}^\Lambda$, we define the vector component-wise operations of addition $\mathbf{c} + \mathbf{c}'$, subtraction $\mathbf{c} - \mathbf{c}'$, and scalar multiplication $n\mathbf{c}$ for $n \in \mathbb{N}$. For a set $\Delta \subset \Lambda$, we view a vector $\mathbf{c} \in \mathbb{N}^\Delta$ equivalently as a vector $\mathbf{c} \in \mathbb{N}^\Lambda$ by assuming $\mathbf{c}(S) = 0$ for all $S \in \Lambda \setminus \Delta$. Write $\mathbf{c} \upharpoonright \Delta$ to denote the vector $\mathbf{d} \in \mathbb{N}^\Delta$ such that $\mathbf{c}(S) = \mathbf{d}(S)$ for all $S \in \Delta$. Given $S_1, \dots, S_k \in \Lambda$, $\mathbf{c} \in \mathbb{N}^\Lambda$, and $n_1, \dots, n_k \in \mathbb{Z}$, we write $\mathbf{c} + \{n_1 S_1, \dots, n_k S_k\}$ to denote vector addition of \mathbf{c} with the vector $\mathbf{v} \in \mathbb{Z}^{\{S_1, \dots, S_k\}}$ with $\mathbf{v}(S_i) = n_i$.

Given a finite set of chemical species Λ , a *reaction* over Λ is a triple $\alpha = \langle \mathbf{r}, \mathbf{p}, k \rangle \in \mathbb{N}^\Lambda \times \mathbb{N}^\Lambda \times \mathbb{R}^+$, specifying the stoichiometry (amount consumed/produced) of the reactants and products, respectively, and the *rate constant* k . A reaction is *unimolecular* if it has one reactant and *bimolecular* if it has two reactants. We use no higher-order reactions in this paper. For simplicity, in this paper we use $k = 1$ and the rate constant is omitted. For instance, given $\Lambda = \{A, B, C\}$, the reaction $A + 2B \rightarrow A + 3C$ is the pair $\langle (1, 2, 0), (1, 0, 3) \rangle$. A (*finite*) *chemical reaction network (CRN)* is a pair $N = (\Lambda, R)$, where Λ is a finite set of chemical *species*, and R is a finite set of reactions over Λ . A *state* of a CRN $N = (\Lambda, R)$ is a vector $\mathbf{c} \in \mathbb{N}^\Lambda$.

Given a state \mathbf{c} and reaction $\alpha = \langle \mathbf{r}, \mathbf{p} \rangle$, we say that α is *applicable* to \mathbf{c} if $\mathbf{r} \leq \mathbf{c}$ (i.e., \mathbf{c} contains enough of each of the reactants for the reaction to occur). If α is applicable to \mathbf{c} , then write $\alpha(\mathbf{c})$

to denote the state $\mathbf{c} + \mathbf{p} - \mathbf{r}$ (i.e., the state that results from applying reaction α to \mathbf{c}). A finite or infinite sequence of reactions (α_i) , where each $\alpha_i \in R$, is a *reaction sequence*. Given an initial state \mathbf{c}_0 and a reaction sequence (α_i) , the induced *execution sequence* (or *path*) q is a finite or infinite sequence of states $q = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \dots)$ such that, for all $\mathbf{c}_i \in q$ ($i \geq 1$), $\mathbf{c}_i = \alpha_i(\mathbf{c}_{i-1})$.⁷ If a finite execution sequence q starts with \mathbf{c} and ends with \mathbf{c}' , we write $\mathbf{c} \Longrightarrow_q \mathbf{c}'$. We write $\mathbf{c} \Longrightarrow \mathbf{c}'$ if such an execution sequence exists and we say that \mathbf{c}' is *reachable* from \mathbf{c} .

3.2 Abstract algebra

A few concepts from abstract algebra have proven useful in describing the reachable states of CRNs, as well as characterizing their computational power (see Section 3.3). A set $A \subseteq \mathbb{N}^k$ is *linear* if $A = \{ \mathbf{b} + \sum_{i=1}^p n_i \mathbf{u}_i \mid n_1, \dots, n_p \in \mathbb{N} \}$ for some constant vectors $\mathbf{b}, \mathbf{u}_1, \dots, \mathbf{u}_p \in \mathbb{N}^k$. A is *semilinear* if it is a finite union of linear sets. A is a *monoid* if $\mathbf{0} \in A$ and $A + A \subseteq A$, i.e., A is closed under addition. We say that A is a *monoid coset* (a.k.a. *monoid offset*) if $A = \mathbf{b} + M$ for some constant vector $\mathbf{b} \in \mathbb{N}^k$ and monoid $M \subseteq \mathbb{N}^k$.

A powerful result due to Leroux helps to restrict the complexity of the set of reachable states. (Leroux actually proves a more powerful result involving the first-order definability of certain sets, but the following implication is sufficient for our purposes.) For any CRN $N = (\Lambda, R)$ and set $X \subseteq \mathbb{N}^\Lambda$, let $\text{post}^N(X) = \{ \mathbf{y} \in \mathbb{N}^\Lambda \mid (\exists \mathbf{x} \in X) \mathbf{x} \Longrightarrow_N \mathbf{y} \}$ be the set of states reachable from some state in X .

Theorem 3.1 ([14]). *If $X \subseteq \mathbb{N}^\Lambda$ is semilinear, then $\text{post}^N(X)$ is a finite union of monoid cosets.*

We will find ourselves frequently dealing with infinite sequences of states. The following technical lemma elucidates certain convenient properties of any such sequence and will be used repeatedly.

Lemma 3.2 (Dickson’s Lemma [10]). *The set of states \mathbb{N}^k is well-quasi-ordered. In particular, every infinite sequence $\mathbf{x}_0, \mathbf{x}_1, \dots$ of states has an infinite nondecreasing subsequence $\mathbf{x}_{i_0} \leq \mathbf{x}_{i_1} \leq \dots$, where $i_0 < i_1 < \dots \in \mathbb{N}$, and every set $U \subseteq \mathbb{N}^k$ has a finite number of minimal elements.*

3.3 Stable decidability of predicates

We now review the definition of stable decidability of predicates introduced by Angluin, Aspnes, and Eisenstat [2]. Intuitively, some species “vote” for a YES/NO answer, and a CRN N is a stable decider if N is guaranteed to reach a consensus vote.

A *chemical reaction decider* (CRD) is a tuple $\mathcal{D} = (\Lambda, R, \Sigma, \Upsilon, \phi, \mathbf{s})$, where (Λ, R) is a CRN, $\Sigma \subseteq \Lambda$ is the *set of input species*, $\Upsilon \subseteq \Lambda$ is the set of *voters*, $\phi : \Upsilon \rightarrow \{\text{NO}, \text{YES}\}$ is the (*Boolean*) *output function*, and $\mathbf{s} \in \mathbb{N}^{\Lambda \setminus \Sigma}$ is the *initial context*. For the input vector $(n_1, \dots, n_k) \in \mathbb{N}^k$, where $k = |\Sigma|$, we write the initial state as $\mathbf{i}(n_1, \dots, n_k) \in \mathbb{N}^\Lambda$ defined by: $\mathbf{i}(n_1, \dots, n_k) \upharpoonright \Sigma = (n_1, \dots, n_k)$ and $\mathbf{i}(n_1, \dots, n_k) \upharpoonright (\Lambda \setminus \Sigma) = \mathbf{s}$. We extend ϕ to a partial function on states $\Psi : \mathbb{N}^\Lambda \dashrightarrow \{\text{NO}, \text{YES}\}$ as follows. $\Psi(\mathbf{c})$ is undefined if either $\mathbf{c}(X) = 0$ for all $X \in \Upsilon$, or if there exist $X_0, X_1 \in \Upsilon$ such that $\mathbf{c}(X_0) > 0$, $\mathbf{c}(X_1) > 0$, $\phi(X_0) = \text{NO}$ and $\phi(X_1) = \text{YES}$. Otherwise, there exists $b \in \{\text{NO}, \text{YES}\}$ such that $(\forall X \in \Upsilon)(\mathbf{c}(X) > 0 \text{ implies } \phi(X) = b)$; in this case, the *output* $\Psi(\mathbf{c})$ of state \mathbf{c} is b .

A state \mathbf{o} is *output stable* if $\Psi(\mathbf{o})$ is defined and, for all \mathbf{c} such that $\mathbf{o} \Longrightarrow \mathbf{c}$, $\Psi(\mathbf{c}) = \Psi(\mathbf{o})$. We call a whole CRD \mathcal{D} *stable* if, for any initial state \mathbf{i} , there exists $b \in \{\text{NO}, \text{YES}\}$ such that, for every state \mathbf{x} reachable from \mathbf{i} , there is an output stable state \mathbf{o} reachable from \mathbf{x} such that $\Psi(\mathbf{o}) = b$.

⁷When the initial state to which a reaction sequence is applied is clear from context, we may overload terminology and refer to a reaction sequence and an execution sequence interchangeably. The possibility that different reactions could result in identical state change (e.g., $A \rightarrow B$ and $A + C \rightarrow B + C$ when C is present) is immaterial to the arguments in this paper.

(i.e., \mathcal{D} always converges to a defined output b on input \mathbf{i} , where b depends only on \mathbf{i} and not on the path taken). If \mathcal{D} is stable, then for some unique subset $S_0 \subseteq \mathbb{N}^k$ of inputs it always converges to output 0 and stays with that output, and for the remainder $S_1 = \mathbb{N}^k \setminus S_0$ it always converges to output 1 and stays with that output. We say that \mathcal{D} *stably decides* the set S_1 , or that \mathcal{D} *stably decides* the predicate $\psi : \mathbb{N}^k \rightarrow \{0, 1\}$ defined by $\psi(\mathbf{x}) = 1$ iff $\mathbf{x} \in S_1$.

A set $A \subseteq \mathbb{N}^k$ is *linear* if $A = \{ \mathbf{b} + \sum_{i=1}^p n_i \mathbf{u}_i \mid n_1, \dots, n_p \in \mathbb{N} \}$ for some constant vectors $\mathbf{b}, \mathbf{u}_1, \dots, \mathbf{u}_p \in \mathbb{N}^k$. A is *semilinear* if it is a finite union of linear sets. The following theorem is due to Angluin, Aspnes, and Eisenstat [2]:

Theorem 3.3 ([2]). *A set $A \subseteq \mathbb{N}^k$ is stably decidable by a CRD if and only if it is semilinear.*

If a YES voter (or any other species, for that matter) cannot be produced by any sequence of reactions from a state \mathbf{y} , then it cannot be produced from any subset $\mathbf{y}' \leq \mathbf{y}$. The following lemma is useful when we want to argue the other way: that for certain species, beyond a certain value, *increasing* their counts cannot affect the ability or inability of the state to produce a YES voter. We say that a state \mathbf{c} is *committed* if, for all states \mathbf{z} such that $\mathbf{c} \implies \mathbf{z}$, $\mathbf{z}(S) = 0$ for all YES-voting species S . In particular, all output-stable NO states are committed, and for stable CRDs, committed states are reachable only from inputs on which the predicate is false.⁸

Lemma 3.4. *For each CRD, there is a constant c such that, for all committed states \mathbf{c} , if $\mathbf{c}(S) > c$ for some $S \in \Lambda$, then for all $n \in \mathbb{Z}$, $\mathbf{c} + \{nS\}$ is also committed.*

Proof. The negation of a state \mathbf{u} being committed is that there exists a (possibly empty) sequence of reactions r such that $\mathbf{u} \implies_r \mathbf{z}$ and there is a YES-voting species Y with $\mathbf{z}(Y) > 0$. Call such a state *uncommitted*, and let U be the set of all uncommitted states. Note that U is “upward closed”: if $\mathbf{u} \in U$, then all $\mathbf{v} \geq \mathbf{u}$ are also uncommitted since r can be applied to \mathbf{v} as well. Hence it is clear that if $n \leq 0$, $\mathbf{c} + \{nS\}$ is committed, since $\mathbf{c} + \{nS\} \leq \mathbf{c}$.

By Dickson’s Lemma, every set $U \subseteq \mathbb{N}^\Lambda$ has a finite number of minimal elements, i.e. vectors $\mathbf{m} \in U$ such that, for all $\mathbf{u} \in U$, $\mathbf{u} \leq \mathbf{m}$ implies $\mathbf{u} = \mathbf{m}$. Call this set of minimal elements M . Suppose for the sake of contradiction that $(\exists \mathbf{u} \in U)(\forall \mathbf{m} \in M) \mathbf{m} \not\leq \mathbf{u}$. Then \mathbf{u} is minimal, hence $\mathbf{u} \in M$, but $\mathbf{u} \leq \mathbf{u}$, which contradicts our supposition.

Thus $(\forall \mathbf{u} \in U)(\exists \mathbf{m} \in M) \mathbf{m} \leq \mathbf{u}$. Because U is upward closed, we conclude that $U = \bigcup_{\mathbf{m} \in M} \{ \mathbf{u} \mid \mathbf{u} \geq \mathbf{m} \}$, i.e., U is a finite union of “cones”. The lemma follows by choosing $c = \max_{\mathbf{m} \in M, S \in \Lambda} \mathbf{m}(S)$. \square

4 Speed fault free CRDs

In this section we show our main result that speed fault free CRDs decide only “detection problems,” i.e., detecting the presence or absence of a species, but not distinguishing between two different positive counts of it. To allow for “parallelization” of the computation, we introduce a “fuel” species F , whose count is allowed to start arbitrarily large.⁹ Increasing the amount of fuel species is analogous to increasing the amount of “receptor” in the introduction. We then formalize the concept of “speed fault free” discussed informally in the introduction. Briefly, a CRN experiences a

⁸A committed state is not be output-stable NO if a state without any voters is reachable from it. The distinct notion of “committed” is useful because (unlike for output NO stability) the negation of committed is closed under superset (see the proof of Lemma 3.4), yet (like for output NO stability) reaching a committed state implies that the predicate value must be false.

⁹Allowing multiple fuel species F_1, F_2, \dots does affect our results since one of our reactions can be $F \rightarrow F_1 + F_2 \dots$

speed fault if it reaches a state from which all paths to a correct state execute some reaction when the counts of all of its reactants are bounded by a constant (a “slow” reaction). Note that in the stochastic model, the expected time for such a reaction to occur is bounded below by a constant (independent of the amount of fuel).

Let $\mathcal{D} = (\Lambda, R, \Sigma, \Upsilon, \phi, \mathbf{s})$ be a stable CRD, where $\Sigma = \{A_1, \dots, A_k\}$ are the input species and $\Lambda \setminus \Sigma$ contains a special “fuel” species F , with variable initial count n . The initial count of every other species in $\Lambda \setminus (\Sigma \cup \{F\})$ is \mathbf{s} (unchanging with respect to n). Write the initial state of \mathcal{D} with some number n_i of each input A_i and n molecules of F as $\mathbf{i}_n(n_1, \dots, n_k)$.

Let $f \in \mathbb{N}$, let $\alpha \in R$ be a reaction and $\mathbf{x} \in \mathbb{N}^\Lambda$ be a state. We say that α occurring in state \mathbf{x} is f -fast if at least one reactant has count at least f in \mathbf{x} . An execution sequence is called f -fast if all reactions in it are f -fast. ¹⁰

Definition 4.1. A stable CRD \mathcal{D} is *speed fault free* if for all n_1, \dots, n_k and all $f \in \mathbb{N}$, for all sufficiently large n ,¹¹ for any state \mathbf{x} such that $\mathbf{i}_n(n_1, \dots, n_k) \Longrightarrow \mathbf{x}$, there is an output stable state \mathbf{y} (which has the correct answer with respect to n_1, \dots, n_k by the stability of \mathcal{D}) such that $\mathbf{x} \Longrightarrow \mathbf{y}$ by an f -fast execution sequence.

In other words, from any reachable state (whether reachable by a fast or slow sequence of reactions), there is always a sequence of fast reactions that reaches the correct answer.

Definition 4.2. A set $S \subseteq \mathbb{N}^k$ is a *simple detection set* if there is a $1 \leq i \leq k$ such that $S = \{ (x_1, \dots, x_k) \in \mathbb{N}^k \mid x_i > 0 \}$. A set is a *detection set* if it is expressible as a combination of finite unions, intersections, and complements of simple detection sets.

In other words, the predicate corresponding to a simple detection set S is a finite Boolean combination of questions of the form “is a certain species present?”. The following theorem is the main result of this paper. We show each direction in two separate lemmas, Lemma 4.4 and Lemma 4.13.

Theorem 4.3. *The sets decidable by speed fault free CRDs are precisely the detection sets.*

4.1 Detection problems are decidable by speed fault free CRDs

Since we argue that the CRD is fast under the standard stochastic time model [12] in addition to being speed fault free, we must first define this model. Since all rate constants in this paper are 1, we define time assuming this to be true.

Given a fixed volume v and current state \mathbf{c} , the *propensity* of a unimolecular reaction $\alpha : X \rightarrow \dots$ in state \mathbf{c} is $\rho(\mathbf{c}, \alpha) = \mathbf{c}(X)$. The propensity of a bimolecular reaction $\alpha : X + Y \rightarrow \dots$, where $X \neq Y$, is $\rho(\mathbf{c}, \alpha) = \frac{\mathbf{c}(X)\mathbf{c}(Y)}{v}$. The propensity of a bimolecular reaction $\alpha : X + X \rightarrow \dots$ is $\rho(\mathbf{c}, \alpha) = \frac{1}{2} \frac{\mathbf{c}(X)(\mathbf{c}(X)-1)}{v}$. The propensity function determines the kinetics of the system as follows. The time until the next reaction occurs is an exponential random variable with rate $\rho(\mathbf{c}) = \sum_{\alpha \in R} \rho(\mathbf{c}, \alpha)$.

The probability that next reaction will be a particular α_{next} is $\frac{\rho(\mathbf{c}, \alpha_{\text{next}})}{\rho(\mathbf{c})}$.

The kinetic model is based on the physical assumption of well-mixedness valid in a dilute solution. Thus, we assume the *finite density constraint*, which stipulates that the volume to execute a CRN must be proportional to the maximum molecular count obtained during execution [16].

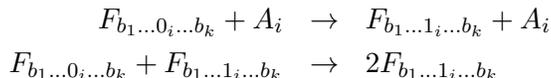
¹⁰It is worth noting that fast reaction sequences are not necessarily fast in the standard sense of stochastic kinetics, since although each reaction occurs quickly, it could be that there are a huge number of reactions in the sequence. Since our main result is a lower bound, this does not hurt the argument (and our upper bound result also shows that it is possible to decide detection problems quickly under the standard stochastic model).

¹¹ n may need to be larger for different input values n_1, \dots, n_k and different constants f .

Lemma 4.4. *Every detection set is decidable by a speed fault free CRD. This CRD takes expected time $O(\log n/n)$ expected time to stabilize under the standard model of stochastic chemical kinetics with constant volume.*

Proof. We describe the CRD using the language of “agent states” from population protocols. By this we mean that for each “agent” F with l different states, there are species F_1, \dots, F_l , and an agent “changing state” from i to j upon reacting with a molecule X means that we have a reaction of the form $X + F_i \rightarrow F_j + \dots$

Let A_1, \dots, A_k the input species of the detection problem. F has 2^k states to keep track of the k bits “is A_i present?” for each $1 \leq i \leq k$, so that each fuel species is written $F_{b_1 \dots b_k}$ where each $b_i \in \{0_i, 1_i\}$, and the initial state of all n F ’s is $F_{0_1 \dots 0_k}$. For each $F_{b_1 \dots 0_i \dots b_k}$, we have the reactions



First we argue that the expected time to converge to an answer is $O(\log n)$. Each F species votes in accordance with the detection problem, according to whether its state indicates the answer is yes or no. We start with n copies of F and at most $O(1)$ copies of each A_i .¹² In expected time $\log k = O(1)$ with respect to n (since k is a constant depending on the detection problem but not on n), each A_i species with positive count has encountered at least one F molecule. It takes $O(\log n)$ expected time for a single bit to propagate to the entire population of F ’s. Therefore it takes at most $kO(\log n) = O(\log n)$ (again, since k is a constant independent of n) for all bits to propagate. At this point all fuel agents have the same state, so they all vote unanimously and correctly.

Now we argue that the CRD is speed fault free. Let \mathbf{x} be any reachable state, and let A_i be a species with $\mathbf{x}(A_i) > 0$ that has not yet reacted with any fuel. Then all n fuel molecules have bit $b_i = 0_i$, so for at least one assignment to the other $k - 1$ bits, some fuel species has count at least $\frac{n}{2^{k-1}}$, so the first reaction above with that fuel species is $\frac{n}{2^{k-1}}$ -fast. Suppose that $\mathbf{x}(A_i) > 0$ but has reacted with at least one fuel. Then the second reaction, executed at most n times, will stabilize the value of bit b_i to be 1_i . Since the number of fuel molecules with bit 0_i plus the number with bit 1_i is exactly n , by a similar argument, there must be some assignment to all k bits such that a fuel species with that assignment has count at least $\frac{n}{2^k}$, whence the second reaction with that species is $\frac{n}{2^k}$ -fast. Therefore there is always a $\frac{n}{2^k}$ -fast path to an output-stable state with the correct answer, which occurs when all bits b_i corresponding to positive-count species A_i become 1_i , implying that the CRD is speed fault free. \square

4.2 Speed fault free CRDs decide only detection problems

Before proceeding to the main argument, we need to develop some technical machinery. We first show that if a fast execution sequence is used to decrease the count of some species, then we can identify certain reactions that must necessarily occur (reaction extraction). We then develop a notion of pumping, which is used to identify species that can get arbitrarily large with increasing fuel. Finally, we show that reaction sequences in which one reactant is always pumpable can be decomposed into separate “test-tubes” (parallel decomposition). Finally we stitch these notions together to show that speed fault free CRDs cannot compute more than detection problems.

¹²So that the total volume required is $O(n)$, although the initial count of A_i maybe be larger than n , we measure time asymptotically with respect to increasing n , so we consider the initial count of A_i to be constant with respect to n . With some care we could handle cases when the count of the A ’s is much larger than n , simply by having all A molecules act like fuels as well, but since this complicates the exposition without changing the basic proof idea, we keep the presentation simply by assuming $n = \Omega(\#_0 A_i)$ for each i .

4.2.1 Reaction extraction lemma

The following lemma is a key technical tool used in our proof of the main result. Intuitively, the lemma below states that a fast reaction sequence that decreases certain species from high counts to low counts must contain reactions of a certain restricted form. These reactions will later be used to do “surgery” on fast reaction sequences, because they give a way to alter the count of certain species, by inserting or removing those reactions, while carefully controlling the effect these insertions and removals have on counts of other species.

Lemma 4.5. *Let $c_1, c_2 \in \mathbb{N}$ such that $c_2 > |\Lambda| \cdot c_1$, let $\mathbf{x}, \mathbf{y} \in \mathbb{N}^\Lambda$ such that $\mathbf{x} \Longrightarrow \mathbf{y}$ via c_2 -fast reaction sequence q . Define $\Delta = \{ D \in \Lambda \mid \mathbf{x}(D) \geq c_2 \text{ and } \mathbf{y}(D) \leq c_1 \}$. Then there is an order on Δ , so that we may write $\Delta = \{D_1, D_2, \dots, D_l\}$, such that, for all $i \in \{1, \dots, l\}$, there is a reaction α_i of the form $D_i \rightarrow P_1 + \dots + P_k$ or $D_i + S \rightarrow P_1 + \dots + P_k$, such that $S, P_1, \dots, P_k \notin \{D_1, \dots, D_i\}$, and α_i occurs at least $\frac{c_2 - |\Lambda| \cdot c_1}{|R|}$ times in q in states \mathbf{c} in which $\mathbf{c}(S) \geq c_2$.*

Proof. We define the ordering based on increasing sets $\emptyset = \Delta_0 \subset \Delta_1 \subset \Delta_2 \subset \dots \Delta_{l-1} \subset \Delta_l = \Delta$, where for each $1 \leq i \leq l$, $\Delta_i \setminus \Delta_{i-1} = \{D_i\}$.

We define the ordering inductively “in reverse,” by first defining D_l , then D_{l-1} , etc. For all $1 \leq i \leq l$, define $\Phi_i : \mathbb{N}^\Lambda \rightarrow \mathbb{N}$ for all states \mathbf{c} by $\Phi_i(\mathbf{c}) = \sum_{D \in \Delta_i} \mathbf{c}(D)$. Φ_l is well-defined since $\Delta_l = \Delta$, and Φ_i is well-defined once we have defined D_{i+1}, \dots, D_l , because $\Delta_i = \Delta \setminus \{D_{i+1}, \dots, D_l\}$.

Because $\mathbf{y}(D) \leq c_1$ for all $D \in \Delta$, it follows that $\Phi_i(\mathbf{y}) \leq i \cdot c_1 \leq |\Lambda| \cdot c_1$. Recall that $\mathbf{x}(D) \geq c_2$ for all $D \in \Delta$. Let r be the suffix of q after the last state \mathbf{c}' along q such that $\Phi_i(\mathbf{c}') \geq c_2$. Then in all states \mathbf{c} in r (not including \mathbf{c}' itself), $\mathbf{c}(D) < c_2$ for all $D \in \Delta_i$. Because $\Phi_i(\mathbf{c}') \geq c_2$, r must contain a subsequence s of reactions, each of which strictly decreases Φ_i , and the total decrease in Φ_i over all of s is at least $(c_2 - |\Lambda| \cdot c_1)$ between states \mathbf{c}' and \mathbf{y} .

Since all reactions in s strictly decrease Φ_i , all reactions must have a reactant in Δ_i . Since s is c_2 -fast, and all states \mathbf{c} along s have $\mathbf{c}(D) < c_2$ for $D \in \Delta_i$, the reaction cannot be unimolecular since the count of D is too low for the reaction to be c_2 -fast, so the reaction must be bimolecular with the other reactant S having count at least c_2 . This implies $S \notin \Delta_i$ (since all $D \in \Delta_i$ have count $< c_2$ between \mathbf{c}' and \mathbf{y}). For the reaction to strictly decrease Φ_i , all products $P \notin \Delta_i$ (otherwise Φ_i would either stay equal or increase after applying the reaction). In fact, this implies every reaction in s decreases Φ_i by exactly 1. Since there are at least $c_2 - |\Lambda| \cdot c_1$ instances of such reactions in s , and there are at most $|R|$ total types of reactions, by the pigeonhole principle at least one reaction type must repeat in s at least $\frac{c_2 - |\Lambda| \cdot c_1}{|R|}$ times. \square

4.2.2 Pumpable sets of species

This section defines *pumpable* sets of species: species whose counts can be made arbitrarily large by increasing the amount of fuel (species F , see Definition 4.1) and proves some basic properties about them. For example, the fuel species F is trivially pumpable. If there is a reaction $F + A \rightarrow F' + A$, then F' is pumpable (if there is an A), because F can be arbitrarily large. To get a handle on the notion of speed fault free, we define pumping to enforce a certain kind of self-consistency (Π -friendly): you can pump without requiring any reactions where all reactants are not pumpable.

Let $\Pi \subseteq \Lambda$. If a reaction has at least one reactant in Π , say the reaction is Π -friendly. If $\mathbf{x} \Longrightarrow \mathbf{y}$ via a reaction sequence in which all reactions are Π -friendly, then we write $\mathbf{x} \Longrightarrow^\Pi \mathbf{y}$. Let $Z = (\mathbf{z}_1 \leq \mathbf{z}_2 \leq \mathbf{z}_3 \dots)$, where each $\mathbf{z}_n \in \mathbb{N}^\Lambda$, be an infinite nondecreasing sequence of states. A set of species $\Pi \subseteq \Lambda$ is *Z-pumpable* if there exists a sequence of states $X = (\mathbf{x}_1, \mathbf{x}_2, \dots)$ such that: (1) for all $P \in \Pi$ and $m \in \mathbb{N}$, $\mathbf{x}_m(P) \geq m$, and (2) for all $m \in \mathbb{N}$, there exists $n \in \mathbb{N}$ such that

$\mathbf{z}_n \Longrightarrow^{\Pi} \mathbf{x}_m$.¹³ Call such a sequence (\mathbf{x}_m) a *pumping sequence* for Π . Π is *maximal Z-pumpable* if it is Z-pumpable and no strict superset of Π is Z-pumpable.

The next proposition shows that after pumping a maximal Π , all other species have bounded counts in all states reachable by Π -friendly paths.

Proposition 4.6. *Let $Z = (\mathbf{z}_1 \leq \mathbf{z}_2 \leq \dots)$ be a infinite nondecreasing sequence of states, and let $\Pi \subseteq \Lambda$ be maximal Z-pumpable, with pumping sequence (\mathbf{x}_m) . Then there is a constant c such that, for all states \mathbf{y} and $m, n \in \mathbb{N}$ such that $\mathbf{x}_m \Longrightarrow^{\Pi} \mathbf{y}$, then for all $S \in \Lambda \setminus \Pi$, $\mathbf{y}(S) < c$.*

Proof. Suppose otherwise. There exists $S \in \Lambda \setminus \Pi$ and an infinite number of $m \in \mathbb{N}$ such that $\mathbf{x}_m \Longrightarrow_{r_m}^{\Pi} \mathbf{y}_m$ and $\mathbf{y}_m(S) \geq k_m$, where k_m increases without bound as $m \rightarrow \infty$. By Dickson’s Lemma we can assume each $\mathbf{x}_m < \mathbf{x}_{m+1}$, so that r_m is applicable to $\mathbf{x}_{m'}$ for all $m' > m$. Then by choosing m' sufficiently large, $\mathbf{x}_{m'} \Longrightarrow_{r_m}^{\Pi} \mathbf{y}_{m'}$, where $\mathbf{y}_{m'}(S) \geq m$ and for all $X \in \Pi$, $\mathbf{y}_{m'}(X) \geq m'$, showing that $\Pi \cup \{S\}$ is Z-pumpable and contradicting the maximality of Π . \square

We will use Proposition 4.6 repeatedly, but its most important consequence, intuitively, is that that the only way to get something outside of Π “large” is by executing a “slow” reaction (between two reactants not in Π). This is captured by the following corollary.

Corollary 4.7. *Let $Z = (\mathbf{z}_1 \leq \mathbf{z}_2 \leq \dots)$ be a infinite nondecreasing sequence of states, and let $\Pi \subseteq \Lambda$ be maximal Z-pumpable, with pumping sequence (\mathbf{x}_m) . Let $c \in \mathbb{N}$ be the constant in Proposition 4.6 ensuring that in any state \mathbf{c} such that $\mathbf{x}_m \Longrightarrow^{\Pi} \mathbf{c}$, $\mathbf{c}(S) < c$ for all $S \in \Lambda \setminus \Pi$. Then if $\mathbf{x}_m \Longrightarrow_p \mathbf{c}'$ and $\mathbf{c}'(S) \geq c$, where $S \in \Lambda \setminus \Pi$, some reaction in p is not c-fast.*

Proof. Proposition 4.6 ensures that if p is Π -friendly, then no such $S \in \Lambda \setminus \Pi$ exists. Therefore p is not Π -friendly. Let α be the first reaction along p that is not Π -friendly. Since the state immediately preceding this reaction is reachable by a Π -friendly path, Proposition 4.6 tells us that all species $S \in \Lambda \setminus \Pi$ have count less than c . Therefore α occurs when the count of all its reactants is less than c , hence it is not c-fast. \square

Proposition 4.8. *Let $Z = (\mathbf{z}_1 \leq \mathbf{z}_2 \leq \dots)$ be a infinite nondecreasing sequence of states, and let $\Pi \subseteq \Lambda$ be maximal Z-pumpable. If R contains a reaction with all reactants in Π , then all products are in Π .*

Proof. For each $m \in \mathbb{N}$, let \mathbf{x}_m be reachable from some \mathbf{z}_n such that $\mathbf{x}_m(S) \geq m$ for all $S \in \Pi$. If $X \neq Y$, then from state \mathbf{x}_m execute the reaction $X + Y \rightarrow \dots$ $m/2$ times. This results in a state in which all products of the reaction, as well as X and Y , have count at least $m/2$, and all other species in Π besides X and Y have count at least $m > m/2$. Since $m/2$ grows unboundedly, this establishes that the products are maximal Z-pumpable as well. If $X = Y$, then execute the reaction $m/3$ times, which ensures that the counts of X , Y , and all products are at least $m/3$, similarly establishing that the products are pumpable. \square

4.2.3 Parallel decomposition

Intuitively, the following lemma shows that systems reacting by Π -friendly reactions can be effectively decomposed into separate non-interacting “test tubes” (in the context of a large excess of Π).¹⁴

¹³We can assume that $n \rightarrow \infty$ as $m \rightarrow \infty$. This is because (\mathbf{z}_n) is a nondecreasing sequence, and so if $\mathbf{z}_n \Longrightarrow^{\Pi} \mathbf{x}_m$ for some $n, m \in \mathbb{N}$, then for all $n' > n$, there is a superset $\mathbf{x}'_m \geq \mathbf{x}_m$ such that $\mathbf{z}_{n'} \Longrightarrow^{\Pi} \mathbf{x}'_m$, and $\mathbf{x}'_m(S) \geq m$ for all $S \in \Pi$.

¹⁴Note that in this way Π -friendly bimolecular reactions act somewhat analogously to unimolecular reactions: if $\mathbf{x} + \mathbf{y} \Longrightarrow \mathbf{z}$ by a sequence of unimolecular reactions, then $\mathbf{x} \Longrightarrow \mathbf{z}'$ and $\mathbf{y} \Longrightarrow \mathbf{z}''$ such that $\mathbf{z}' + \mathbf{z}'' = \mathbf{z}$.

For a reaction sequence q applied to a state \mathbf{x} to give $\mathbf{x} \Longrightarrow_q \mathbf{y}$, where \mathbf{x} is written as a sum of two states $\mathbf{x}_1 + \mathbf{x}_2 = \mathbf{x}$, we say that q has a *parallel decomposition from* $(\mathbf{x}_1, \mathbf{x}_2)$ if there exists a partition of q into two disjoint subsequences of reactions (q_1, q_2) such that $\mathbf{x}_1 \Longrightarrow_{q_1} \mathbf{y}_1$, $\mathbf{x}_2 \Longrightarrow_{q_2} \mathbf{y}_2$, and $\mathbf{y} = \mathbf{y}_1 + \mathbf{y}_2$. In other words, if we imagine splitting \mathbf{x} into two “tubes” \mathbf{x}_1 and \mathbf{x}_2 , then the evolution determined by the reaction sequence q can be interpreted as happening entirely within the tubes.

Suppose a reaction sequence p is applicable to $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$, but p does not have a parallel decomposition from $(\mathbf{x}_1, \mathbf{x}_2)$. Then there is a longest prefix q of p (possibly q is empty) such that q has a parallel decomposition (q_1, q_2) from $(\mathbf{x}_1, \mathbf{x}_2)$, then we call q the *join* of p from $(\mathbf{x}_1, \mathbf{x}_2)$. Write $(\mathbf{x}'_1, \mathbf{x}'_2)$ to be such that $\mathbf{x}_1 \Longrightarrow_{q_1} \mathbf{x}'_1$ and $\mathbf{x}_2 \Longrightarrow_{q_2} \mathbf{x}'_2$. In other words, q is the furthest that \mathbf{x}_1 and \mathbf{x}_2 can evolve on their own before the next reaction in p requires a molecule from \mathbf{x}'_1 and a molecule from the other \mathbf{x}'_2 . Therefore the next reaction must be bimolecular $X_1 + X_2 \rightarrow \dots$, and it must be the case that $\mathbf{x}'_1(X_2) = 0$ and $\mathbf{x}'_2(X_1) = 0$, otherwise one of the reaction sequences q_1 or q_2 could be extended by that reaction while remaining a parallel decomposition, and q would not be the longest prefix of p with a parallel decomposition.

Lemma 4.9. *Suppose $\mathbf{x} + \mathbf{y} \Longrightarrow^\Pi \mathbf{z}$. Then there are $\mathbf{p}, \mathbf{p}', \mathbf{p}'' \in \mathbb{N}^\Pi$, and $\mathbf{z}', \mathbf{z}'' \in \mathbb{N}^\Lambda$ such that $\mathbf{p} + \mathbf{x} \Longrightarrow^\Pi \mathbf{p}' + \mathbf{z}'$ and $\mathbf{p} + \mathbf{y} \Longrightarrow^\Pi \mathbf{p}'' + \mathbf{z}''$, where $\mathbf{z}' + \mathbf{z}'' = \mathbf{z}$ and $\mathbf{p}' + \mathbf{p}'' = 2\mathbf{p}$.*

Proof. Let p be the Π -friendly reaction sequence $\mathbf{x} + \mathbf{y} \Longrightarrow \mathbf{z}$. Let $\mathbf{p}_n \in \mathbb{N}^\Pi$ consist of exactly n molecules of every species in Π . For any \mathbf{p}_n , consider the path: $2\mathbf{p}_n + \mathbf{x} + \mathbf{y} \Longrightarrow_p 2\mathbf{p}_n + \mathbf{z}$. Let q_n be the join of p from $(\mathbf{p}_n + \mathbf{x}, \mathbf{p}_n + \mathbf{y})$ and let (r_n, l_n) be the parallel decomposition of q from $(\mathbf{p}_n + \mathbf{x}, \mathbf{p}_n + \mathbf{y})$. Let \mathbf{l}_n and \mathbf{r}_n be such that $\mathbf{p}_n + \mathbf{x} \Longrightarrow_{r_n} \mathbf{l}_n$ and $\mathbf{p}_n + \mathbf{y} \Longrightarrow_{l_n} \mathbf{r}_n$. If $q_n = p$ then we are done: $\mathbf{p}_n + \mathbf{x} \Longrightarrow \mathbf{l}_n$ by a Π -friendly reaction sequence and $\mathbf{p}_n + \mathbf{y} \Longrightarrow \mathbf{r}_n$ by a Π -friendly reaction sequence where $\mathbf{l}_n + \mathbf{r}_n = \mathbf{z} + 2\mathbf{p}$. Otherwise, q_n is not all of p . The next reaction in p after q_n must be of the form $L + R \rightarrow \dots$ where, without loss of generality, $\mathbf{l}_n(R) = 0$, $\mathbf{l}_n(L) > 0$ and $\mathbf{r}_n(L) = 0$, $\mathbf{r}_n(R) > 0$. Since p is Π -friendly, at least one reactant L or R is in Π . Now consider q_{n+1} , the join of p from $(\mathbf{p}_{n+1} + \mathbf{x}, \mathbf{p}_{n+1} + \mathbf{y})$. q_{n+1} must be longer by at least 1 reaction by the following argument. If $L \in \Pi$ then $\mathbf{r}_{n+1}(L) = 1$, and if $R \in \Pi$ then $\mathbf{l}_{n+1}(L) = 1$. Since we still have $\mathbf{r}_{n+1}(R) > 0$ and $\mathbf{l}_{n+1}(L) > 0$, the above reaction from p can occur in either l_{n+1} or r_{n+1} . Since p is finite, at some point $q = p$ and we are done. \square

4.2.4 Main proof

Throughout this section, let $\mathcal{D} = (\Lambda, R, \Sigma, \Upsilon, \phi, \mathbf{s})$ be an arbitrary speed fault free CRD with $\Sigma = \{A_1, \dots, A_k\}$ and fuel species F as in Definition 4.1. Supposing for the sake of contradiction that \mathcal{D} decides some non-detection set, then there must exist some species A_i (assume without loss of generality that $i = 1$), and an input value $(n_1, n_2, \dots, n_k) \in \mathbb{N}^k$, where $n_1 \geq 1$, with answer NO (without loss of generality) but input value $(n_1 + 1, n_2, \dots, n_k)$ with answer YES.

For each $n \in \mathbb{N}$, write the initial state of \mathcal{D} with answer NO and n fuels as $\mathbf{i}_n (= \{n_1 A_1, n_2 A_2, \dots, n_k A_k, nF, \mathbf{s} \upharpoonright (\Lambda \setminus (\Sigma \cup \{F\}))\})$. Our argument will focus on the initial state \mathbf{i}_n with answer NO, for sufficiently large n , and \mathbf{i}_n with one extra copy of A_1 , which has answer YES. We will show that for sufficiently large n , $\mathbf{i}_n + \{A_1\}$ is able to reach a state without YES-voting species, from which the only way to produce a YES voter is to execute a slow bimolecular reaction.

Let $I = (\mathbf{i}_1, \mathbf{i}_2, \dots)$ be the infinite increasing sequence of all such initial states with intended answer NO.

We first observe that for a maximal I -pumpable set Π , due to Theorem 3.1, there is a very controlled manner in which the species in Π can be pumped.

Lemma 4.10. *Let Π be maximal I -pumpable. There exists $\mathbf{d} \in \mathbb{N}^\Lambda$, with $\mathbf{d}(S) > 0 \iff S \in \Pi$, and an infinite sequence of states $\mathbf{x}_1, \mathbf{x}_2, \dots$ such that, for all $m \in \mathbb{Z}^+$, $\mathbf{x}_m + \mathbf{d} = \mathbf{x}_{m+1}$, and there exists $n \in \mathbb{N}$ such that $\mathbf{i}_n \implies^\Pi \mathbf{x}_m$.*

Proof. For each $m \in \mathbb{N}$, let \mathbf{x}'_m be reachable via Π -friendly reactions from some \mathbf{i}_n such that $\mathbf{x}'_m(S) \geq m$ for all $S \in \Pi$. Note that I is semilinear (in fact linear). Remove all reactions from our original CRN that are not Π -friendly to obtain a new CRN N_Π . We then apply Theorem 3.1 to the new CRN N_Π to obtain the following. There exists $\mathbf{b}_1, \dots, \mathbf{b}_l \in \mathbb{N}^\Lambda$ and monoids $M_1, \dots, M_l \subseteq \mathbb{N}^\Lambda$ such that $\text{post}^{N_\Pi}(I) = \bigcup_{j=1}^l (\mathbf{b}_j + M_j)$. Let $n > \max_{1 \leq j \leq l, S \in \Lambda} \mathbf{b}_j(S)$. Since $\mathbf{x}'_n \in \text{post}^{N_\Pi}(I)$, there is a j such that $\mathbf{x}'_n \in \mathbf{b}_j + M_j$.

Let $\mathbf{d} = \mathbf{x}'_n - \mathbf{b}_j$. Define $\mathbf{x}_1 = \mathbf{x}'_n$, and for all $m \in \mathbb{Z}^+$, define $\mathbf{x}_{m+1} = \mathbf{x}_m + \mathbf{d}$. Note that each $\mathbf{x}_m \in \mathbf{b}_j + M_j$ since M_j is closed under addition, so \mathbf{x}_m is reachable from sufficiently large \mathbf{i}_n . Since $\mathbf{x}'_n(S) > \mathbf{b}_j(S)$ for all $S \in \Pi$, $\mathbf{d}(S) = \mathbf{x}'_n(S) - \mathbf{b}_j(S) > 0$. The fact that $\mathbf{d}(S) > 0 \implies S \in \Pi$ follows from the maximality of Π : any species S for which $\mathbf{d}(S) > 0$ satisfies $\mathbf{x}_m(S) \geq m$, which implies that $S \in \Pi$. \square

Define the sequence of output-stable NO states (\mathbf{y}_m) inductively as follows. For the base case, let \mathbf{y}_1 be any output-stable NO state such that $\mathbf{x}_1 \implies_{r_1} \mathbf{y}_1$; such a path r_1 must exist because \mathcal{D} is stable. Inductively assume that $\mathbf{x}_{m-1} \implies_{r_{m-1}} \mathbf{y}_{m-1}$. Then $\mathbf{x}_m = \mathbf{x}_{m-1} + \mathbf{d} \implies_{r_{m-1}} \mathbf{y}_{m-1} + \mathbf{d}$. Let $f_m \in \mathbb{N}$ be the largest number such that there is a f_m -fast path p_m from $\mathbf{y}_{m-1} + \mathbf{d}$ to an output-stable NO state \mathbf{y}_m .¹⁵ Then let r_m be r_{m-1} followed by p_m . In other words $r_m = \bigoplus_{i=1}^m p_i$, where \bigoplus denotes concatenation.¹⁶ By Corollary 4.7, once f is sufficiently large, any f -fast reaction sequence from \mathbf{x}_m to \mathbf{y}_m must be Π -friendly. Thus by reindexing (\mathbf{x}_m) to start with a sufficiently large member of the sequence, we have that for all m , $\mathbf{x}_m \implies^\Pi \mathbf{y}_m$.

By Dickson's Lemma there is an infinite nondecreasing subsequence $Y = (\mathbf{y}_{s_1}, \mathbf{y}_{s_2}, \dots)$. Let $\Gamma = \{S \in \Lambda \mid \lim_{n \rightarrow \infty} \mathbf{y}_{s_n}(S) = \infty\}$. By Proposition 4.6, $\Gamma \subseteq \Pi$ since $\mathbf{x}_{s_n} \implies^\Pi \mathbf{y}_{s_n}$. Let $\Delta = \Pi \setminus \Gamma$. These are the species that are “large” in (\mathbf{x}_m) but are bounded in Y . By the definition of Γ , there is a constant c such that, for all $n \in \mathbb{N}$ and $S \in \Delta$, $\mathbf{y}_{s_n}(S) < c$. Thus an infinite subsequence of \mathbf{y}_{s_n} 's has equal counts of all species $S \in \Lambda \setminus \Gamma$. Since counts of species in Γ grow without bound as $n \rightarrow \infty$, there is an infinite subsequence of *that* on which *every* element of Γ has its count strictly increase on each subsequent \mathbf{y}_{s_n} state. We therefore assume that $Y = (\mathbf{y}_{s_1}, \mathbf{y}_{s_2}, \dots)$ is exactly this infinite subsequence (to avoid introducing yet more variable names), where each $\mathbf{y}_{s_n}(S) = \mathbf{y}_{s_{n+1}}(S)$ if $S \in \Lambda \setminus \Gamma$ and $\mathbf{y}_{s_n}(S) < \mathbf{y}_{s_{n+1}}(S)$ if $S \in \Gamma$.

Recall that a state is *committed* if it cannot produce a YES voter. The next lemma shows that changing counts of pumpable species (Π) by a “small” amount in \mathbf{x}_m , so long as m is sufficiently large, cannot change the ability of \mathbf{x}_m to reach a committed state. Intuitively, later on \mathbf{e} will represent a change in counts due to “processing” the extra copy of A_1 (the one that changes the correct answer in state $\mathbf{i}_n(n_1, \dots, n_k)$ from NO to YES), and the following lemma will help us to derive a contradiction because the extra copy of A_1 should enable the production of a YES voter.

Lemma 4.11. *Let sequences (\mathbf{x}_m) and (\mathbf{y}_m) be as defined above. For all $\epsilon \in \mathbb{N}$, there exists $\epsilon' \in \mathbb{N}$ such that the following holds. For all $\mathbf{e} \in \mathbb{Z}^\Pi$ with $|\mathbf{e}| \leq \epsilon$, for infinitely many m , there exists $\mathbf{e}_m \in \mathbb{Z}^\Gamma$ with $|\mathbf{e}_m| \leq \epsilon'$, and $m_2 < m$ such that $\mathbf{x}_m + \mathbf{e} \implies^\Pi \mathbf{y}_{m_2} + \mathbf{e}_m$ and $\mathbf{y}_{m_2} + \mathbf{e}_m$ is committed.*

¹⁵Let $f_m = f_{m-1} + 1$ if there are f -fast paths from $\mathbf{y}_{m-1} + \mathbf{d}$ to \mathbf{y}_m for all $f \in \mathbb{N}$; this may occur if there are reactions such as $X \rightarrow X + F$ that can generate arbitrarily large counts of some molecules without consuming others.

¹⁶By the definition of speed fault free, $\lim_{m \rightarrow \infty} f_m = \infty$, since \mathbf{x}_m and \mathbf{y}_m for increasing m are reachable from input states \mathbf{i}_n with increasing amounts of fuel.

In other words, if we perturb \mathbf{x}_m by a small \mathbf{e} , then it retains the ability to reach a state that is committed (if not output-stable NO), and that state will be “close” to one of the output-stable NO states \mathbf{y}_{m_2} in the sequence Y . Although the exact distance \mathbf{e}_m from \mathbf{y}_{m_2} depends on m , we can place a bound ϵ' on that distance that depends only on ϵ , the size of the perturbation of \mathbf{x}_m .

Proof. We will prove the lemma by using reaction sequences p_i as defined before the lemma, which are Π -friendly, and we will modify them only by inserting or removing reactions already present in the p_i 's; hence they will remain Π -friendly. Therefore for the rest of the proof we do not explicitly annotate reachability relations between states to denote that they are Π -friendly.

Let $m_1, m_2 \in \mathbb{N}$ such that $\mathbf{y}_{m_1}, \mathbf{y}_{m_2} \in Y$. The proof will establish that if we will choose m_1 to be sufficiently large compared to \mathbf{e} and m_2 to be sufficiently large compared to m_1 and \mathbf{e} , then the lemma will hold for $m = 2m_2 - m_1$. In particular, choose m_1 sufficiently large that $f_{m_1} \geq \hat{c} + |\Lambda| \cdot c$, where f_{m_1} and c are defined before the statement of the lemma, and \hat{c} is defined below (but depends only on the CRN and not on c or m_1).

Let $k = m_2 - m_1$ and let $m = m_2 + k = 2m_2 - m_1$. Then defining the reaction sequence $q = \bigoplus_{i=m_1}^{m_2} p_i$, we have that $\mathbf{y}_{m_1} + k\mathbf{d} \xRightarrow{q} \mathbf{y}_{m_2}$ via an f_{m_1} -fast reaction sequence. Recalling the definition of the reaction sequence $r_{m_1} = \bigoplus_{i=1}^{m_1} p_i$, we have that $\mathbf{x}_m = \mathbf{x}_{m_1} + 2k\mathbf{d} \xRightarrow{r_{m_1}} \mathbf{y}_{m_1} + 2k\mathbf{d} \xRightarrow{q} \mathbf{y}_{m_2} + k\mathbf{d}$. Let $\mathbf{d}_y = \mathbf{y}_{m_2} - \mathbf{y}_{m_1}$, noting that $\mathbf{d}_y(S) \geq 0$ for all $S \in \Lambda$ since $\mathbf{y}_{m_1} \leq \mathbf{y}_{m_2}$. Then $\mathbf{y}_{m_2} + k\mathbf{d} = \mathbf{y}_{m_1} + \mathbf{d}_y + k\mathbf{d} \xRightarrow{q} \mathbf{y}_{m_2} + \mathbf{d}_y$. Recall that all states in Y are committed. Further, recall that by the definition of Y , $\mathbf{d}_y(S) > 0$ if and only if $S \in \Gamma$, since all states in Y have equal counts on species in $\Lambda \setminus \Gamma$. Therefore by Lemma 3.4 (choosing m_2 sufficiently large that $\mathbf{y}_{m_2}(S) > c$ for all $S \in \Gamma$, where c is the constant in Lemma 3.4), $\mathbf{y}_{m_2} + \mathbf{d}_y$ is committed.

Now consider the state $\mathbf{x}_m + \mathbf{e}$. Choosing k large enough compared to ϵ , we have $\mathbf{x}_m + \mathbf{e} \xRightarrow{r_{m_1}} \mathbf{y}_{m_1} + 2k\mathbf{d} + \mathbf{e}$. (We can think of r_{m_1} processing the subset $\mathbf{x}_{m_1} < \mathbf{x}_m$ first, leaving the molecules represented by $2k\mathbf{d}$ alone until \mathbf{y}_{m_1} is reached from \mathbf{x}_{m_1} .)

Then consider applying q to $\mathbf{y}_{m_1} + 2k\mathbf{d} + \mathbf{e}$. We have $\mathbf{y}_{m_1} + 2k\mathbf{d} + \mathbf{e} \xRightarrow{q} \mathbf{y}_{m_2} + k\mathbf{d} + \mathbf{e}$. Since k was chosen larger than the absolute value of any element of \mathbf{e} , this is a valid reaction pathway. The key idea is that we now will do surgery on q to create a new reaction sequence q' , such that $\mathbf{y}_{m_1} + 2k\mathbf{d} + \mathbf{e} \xRightarrow{q'} \mathbf{y}_{m_2} + k\mathbf{d} + \mathbf{e}'$, where $\mathbf{e}' \in \mathbb{Z}^\Gamma$. That is, we show how to alter q through the addition or removal of reactions so that elements of $\Lambda \setminus \Gamma$ end up with exactly the counts they had in $\mathbf{y}_{m_2} + k\mathbf{d}$.

Since $\mathbf{d}_y \in \mathbb{N}^\Gamma$ and $\mathbf{e}' \in \mathbb{Z}^\Gamma$, and each component of \mathbf{d}_y increases unboundedly with k , we can ensure that $\mathbf{d}_y + \mathbf{e}' \geq 0$ on all elements by increasing k . (As we'll see below, \mathbf{e}' can be globally bounded as a function of \mathbf{e} and the CRN, and thus independently of m_1 and m_2 . Thus, $|\mathbf{e}'|$ is not unboundedly increasing with k and we avoid circularity.) Since $\mathbf{d}_y + \mathbf{e}' \geq 0$, we can apply q to $\mathbf{y}_{m_2} + k\mathbf{d} + \mathbf{e}'$, i.e., $\mathbf{y}_{m_2} + k\mathbf{d} + \mathbf{e}' = \mathbf{y}_{m_1} + \mathbf{d}_y + k\mathbf{d} + \mathbf{e}' \xRightarrow{q} \mathbf{y}_{m_2} + \mathbf{d}_y + \mathbf{e}'$. Then Lemma 3.4 implies that the resulting state $\mathbf{y}'_m = \mathbf{y}_{m_2} + \mathbf{d}_y + \mathbf{e}'$ is committed. Taking $\mathbf{e}_m = \mathbf{d}_y + \mathbf{e}'$ satisfies the statement of the lemma. Note that \mathbf{e}_m depends on m because \mathbf{d}_y does.

We now describe how we alter the reaction sequence q applied on $\mathbf{y}_{m_1} + 2k\mathbf{d} + \mathbf{e}$ to achieve these properties.

By Lemma 4.5 (taking $\mathbf{x} = \mathbf{y}_{m_1} + k\mathbf{d}$, $\mathbf{y} = \mathbf{y}_{m_2}$, $c_1 = \max_{D \in \Delta} \mathbf{y}_{m_2}(D)$, $c_2 = |\Lambda| \cdot c_1 + |R|(c_s + 1)^l c_0$, where c_s is the maximum stoichiometric coefficient of any reaction, $c_0 = \max_{S \in \Pi} |\mathbf{e}(S)|$, and $|R|$ is the number of reactions in the CRN), there is an ordering on Δ , so that we can write $\Delta = \{D_1, \dots, D_l\}$, such that for each $1 \leq i \leq l$, there is a reaction α_i with the following properties: 1) α_i has D_i as a reactant. 2) All products and the other reactant are either elements of Γ , or are D_j for $j > i$. The fact that the other reactant must be in Π follows from the fact that Lemma 4.5 establishes its count reaches at least c_2 , so Proposition 4.6 implies it is in Π . The fact that both reactants are

in Π implies all products are in Π by Proposition 4.8. 3) α_i occurs at least $\frac{c_2 - |\Lambda| \cdot c_1}{|R|} = (c_s + 1)^l c_0$ times in q .

We observe that it gives a way to “fix” q to ensure that the counts of species in $\Lambda \setminus \Gamma$ will be zero in \mathbf{e}' (at the end of the proof we argue that $(c_s + 1)^l c_0$ is sufficiently many reactions of the form given by Lemma 4.5 to do this fixing). We iteratively fix the counts of species in Δ one by one, in the ordering given, i.e. we first adjust q to fix D_1 , then we fix D_2 (while showing that the fixing of D_2 cannot affect the count of D_1 in any state, so it remains fixed), etc. We can define a sequence of $\mathbf{e}'_i \in \mathbb{Z}^{\Gamma \cup \{D_{i+1}, \dots, D_l\}}$ to represent the difference in counts compared to $\mathbf{y}_{m_2} + k\mathbf{d}$ resulting from fixing only the first i species D_1, \dots, D_i (with $\mathbf{e}'_0 = \mathbf{e}$, and $\mathbf{e}'_l = \mathbf{e}'$). The strategy for each i is this: If D_i is negative in \mathbf{e}'_{i-1} , then we remove enough instances of α_i from q ; property (3) ensures that q contains enough instances of α_i . If on the other hand D_i 's count in \mathbf{e}'_{i-1} is positive, then we add instances of α_i at the start of q . By property (2), adding or removing instances of α_i affects only the counts of species in Γ and D_{i+1}, \dots, D_l . Since we fix these counts in the prescribed order, when we are done, the counts of each D_i is equal to its count in $\mathbf{y}_{m_2} + k\mathbf{d}$, while counts of elements of Γ have been altered. Now, the altered reaction sequence q' might take species in $\Gamma \cup \Delta$ to counts lower than in the unaltered execution sequence traversed via q . However, as before, this does not cause a problem because we can arbitrarily increase k . Although q will change with k , circularity is avoided because there is a bound on the number of reaction instances added and removed as a function of \mathbf{e} and the CRN, independent of k .

Bound on the amount of fixing: Let $c_0 = \max_{S \in \Pi} |\mathbf{e}(S)|$ and let c_s be the maximum stoichiometry coefficient of any product (which bounds the amount that species can change each time the reaction is added or removed). Then we add or remove at most $\epsilon_1 = c_0$ instances of α_1 , which affects the count of D_2, \dots, D_l and species in Γ by at most $c_0 c_s$. Thus, $|\mathbf{e}'_1| \leq (1 + c_s)c_0$ (the original c_0 error plus the additional $c_0 c_s$ error from altering the number of α_1 reactions). At this point, in the worst case, the value of D_2 in \mathbf{e}'_1 is $(1 + c_s)c_0$, and we have to add or remove $\epsilon_2 = (1 + c_s)c_0$ instances of α_2 , resulting in $|\mathbf{e}'_2| \leq (1 + c_s)^2 c_0$. In general, $\epsilon_i = (1 + c_s)^{i-1} c_0$, and $\mathbf{e}'_i \leq (1 + c_s)^i c_0$. This implies that $\sum_{i=1}^l \epsilon_i = ((1 + c_s)^l - 1)c_0/c_s$ is an upper bound on the total number of reaction instances we add or remove, and $|\mathbf{e}'_l| = |\mathbf{e}'| \leq (c_s + 1)^l c_0$ is an upper bound on the species in Γ we generate or consume when fixing species in Δ . \square

The next lemma uses Lemma 4.11 to show that, from state $\mathbf{x}_m + \mathbf{e}$, with $\mathbf{e} \in \mathbb{Z}^\Lambda$ “small,” we can reach a committed state in which every species that can be “large”, is actually large.

Lemma 4.12. *Let sequence (\mathbf{x}_m) be as defined above. For all $\epsilon \in \mathbb{N}$, there exists $c \in \mathbb{N}$ and $\Omega \subseteq \Lambda$ such that the following holds. For all $\mathbf{e} \in \mathbb{Z}^\Pi$ such that $|\mathbf{e}| \leq \epsilon$, there exists an infinite sequence $W_{\mathbf{e}} = (\mathbf{w}_n)$ of states such that, for all $n \in \mathbb{N}$, there exists $m_n \in \mathbb{N}$, such that the following is true: (1) $\mathbf{x}_{m_n} + \mathbf{e} \Longrightarrow^\Pi \mathbf{w}_n$, (2) \mathbf{w}_n is committed, (3) for all $S \in \Omega$, $\mathbf{w}_n(S) \geq n$, (4) for all $S \in \Lambda \setminus \Omega$ and all \mathbf{u} such that $\mathbf{w}_n \Longrightarrow^\Omega \mathbf{u}$, $\mathbf{u}(S) \leq c$, and (5) \mathbf{w}_n are nondecreasing.*

Proof. Choose ϵ' for ϵ as in Lemma 4.11. From Lemma 4.11, recall the infinite sequence $(\mathbf{y}_{m_2} + \mathbf{e}_m)$, where m ranges over an infinite number of values, and for each m , $m_2 < m$ is a function of m . Define the constant $\mathbf{h} \in \mathbb{N}^\Gamma$ by $\mathbf{h}(S) = \epsilon'$ for all $S \in \Gamma$. From the infinite sequence of m 's, define the infinite sequence $(\mathbf{y}_{m_2} - \mathbf{h})$, letting the k 'th element of this sequence be denoted \mathbf{z}_k , and let $Z = (\mathbf{z}_k)$ denote this sequence. Note that $\mathbf{y}_{m_2} - \mathbf{h} \leq \mathbf{y}_{m_2} + \mathbf{e}_m$, since ϵ' bounds the absolute value of any component of \mathbf{e}_m .

Let Ω be a maximal Z -pumpable set, and let (\mathbf{w}'_n) be a pumping sequence for Ω . Because each $\mathbf{z}_k \leq \mathbf{y}_{m_2}$, and $\mathbf{x}_m \Longrightarrow^\Pi \mathbf{y}_{m_2}$, we have that $\Omega \subseteq \Pi$ by Proposition 4.6. By the fact that each \mathbf{z}_k is \mathbf{y}_{m_2} minus a constant \mathbf{h} for arbitrarily large m_2 , for all $S \in \Gamma$, $\lim_{k \rightarrow \infty} \mathbf{z}_k(S) = \infty$. Since Ω

is Z -maximal, any species S that gets arbitrarily large in Z must be in Ω (if $\mathbf{z}_k \xRightarrow[p_n]{\Omega} \mathbf{w}'_n$ then a superset of \mathbf{w}'_n with arbitrarily much S can be reached from $\mathbf{z}_{k'}$ for larger $k' > k$ by taking p_n). Thus $\Gamma \subseteq \Omega$. Intuitively, Γ contains only species that happened to grow without bound in the states \mathbf{y}_m , whereas Ω contains all those species that, starting from (a state close to) \mathbf{y}_m , *could* grow without bound. Recall (\mathbf{w}'_n) is a pumping sequence for Ω , i.e., for all $n \in \mathbb{N}$, there exists $k \in \mathbb{N}$ such that $\mathbf{z}_k \xRightarrow{\Omega} \mathbf{w}'_n$ and $\mathbf{w}'_n(S) \geq n$ for all $S \in \Omega$. For each \mathbf{e} , for all n , define $\mathbf{w}_n = \mathbf{w}'_n + \mathbf{h} - \mathbf{e}_{m_n}$, where m_n is the value of m corresponding to k (recall to each \mathbf{z}_k there corresponds an m). Thus for each \mathbf{e} we have two infinite sequences \mathbf{w}_n and \mathbf{w}'_n . For each \mathbf{e} , using Dickson's Lemma twice, consider the infinite subsequence of n 's in which both \mathbf{w}_n and \mathbf{w}'_n are nondecreasing. To avoid introducing new variables, consider \mathbf{w}_n and \mathbf{w}'_n to be exactly the nondecreasing subsequences.

We now show that (\mathbf{w}_n) has the properties claimed in the statement of the lemma. By definition of pumping sequence, for all n , there is some path $\mathbf{z}_k \xRightarrow[p_n]{\Omega} \mathbf{w}'_n$. Since $\mathbf{z}_k = \mathbf{y}_{m_2} - \mathbf{h}$ and $\Omega \subseteq \Pi$, this implies $\mathbf{y}_{m_2} - \mathbf{h} \xRightarrow[p_n]{\Pi} \mathbf{w}'_n$. Since $\mathbf{x}_{m_n} + \mathbf{e} \xRightarrow{\Pi} \mathbf{y}_{m_2} + \mathbf{e}_{m_n} \geq \mathbf{y}_{m_2} - \mathbf{h}$, p_n is applicable to $\mathbf{y}_{m_2} + \mathbf{e}_{m_n}$. Therefore $\mathbf{x}_{m_n} + \mathbf{e} \xRightarrow{\Pi} \mathbf{y}_{m_2} + \mathbf{e}_{m_n} \xRightarrow[p_n]{\Pi} \mathbf{w}'_n + \mathbf{h} - \mathbf{e}_{m_n} = \mathbf{w}_n$, so (1) holds. Since all states reachable from a committed state are committed, and $\mathbf{y}_{m_2} + \mathbf{e}_{m_n}$ is committed, it follows that \mathbf{w}_n is committed, so (2) holds. Since (\mathbf{w}'_n) is a pumping sequence for Ω , $\mathbf{w}'_n(S) \geq n$ for all $S \in \Omega$, so (3) follows from the fact that $\mathbf{w}_n \geq \mathbf{w}'_n$. Since \mathbf{w}'_n are increasing in Ω and are nondecreasing in other species, and $\mathbf{h} - \mathbf{e}_{m_n}$ is over $\Gamma \subseteq \Omega$, for each n , $\mathbf{w}'_n \leq \mathbf{w}_n \leq \mathbf{w}'_{n'}$ for some $n' > n$. Thus, since Ω is maximal, (4) follows by Proposition 4.6. Finally, recall that for each \mathbf{e} , $W_{\mathbf{e}} = (\mathbf{w}_n)$ is nondecreasing. \square

The next lemma shows the converse of Lemma 4.4, that speed fault free CRDs decide only detection problems.

Lemma 4.13. *\mathcal{D} is not speed fault free.*

Proof. For the sake of contradiction suppose otherwise. Let n_0 be the constant of Lemma 3.4. Let $\mathbf{e} = \mathbf{0}$ and consider the corresponding $W_{\mathbf{e}} = (\mathbf{w}_n)$. Take $n > n_0$; then since \mathbf{w}_n is committed, it remains committed even when arbitrary counts of species in Ω are added to it (where Ω is the set of species from Lemma 4.12 whose count is at least n in \mathbf{w}_n).

Now, for n' and m corresponding to n , we have $\mathbf{i}_{n'} \xRightarrow{\Pi} \mathbf{x}_m \xRightarrow{\Pi} \mathbf{w}_n$. We can rewrite this path as $(\mathbf{i}_{n'} \setminus \{A_1\}) + \{A_1\} \xRightarrow{\Pi} \mathbf{w}_n$, and applying Lemma 4.9 obtain that there are $\mathbf{p}, \mathbf{p}' \in \mathbb{N}^{\Pi}$ and $\mathbf{b} \in \mathbb{N}^{\Lambda}$ where $\mathbf{b} \leq \mathbf{w}_n$ such that: $\mathbf{p} + \{A_1\} \xRightarrow{\Pi} \mathbf{p}' + \mathbf{b}$. Call this path r . Since $\mathbf{b} \leq \mathbf{w}_n$, it must be that \mathbf{b} is committed even if any amount of Ω is added to it.

Let $\mathbf{e} = \mathbf{p}' - \mathbf{p} \in \mathbb{Z}^{\Pi}$ and consider the corresponding $W_{\mathbf{e}} = (\mathbf{w}_n)$. For all n , by Lemma 4.12, there is m such that $\mathbf{x}_m + \mathbf{e} \xRightarrow{\Pi} \mathbf{w}_n$ by some path p_n . Now, choose n large enough that $n > n_0$ as well as $\mathbf{x}_m \geq \mathbf{p}$. Then, adding *an extra molecule* of A_1 , $\mathbf{x}_m + \{A_1\} \xRightarrow[r]{\Pi} \mathbf{x}_m + (\mathbf{p}' - \mathbf{p}) + \mathbf{b} = \mathbf{x}_m + \mathbf{e} + \mathbf{b} \xRightarrow[p_n]{\Pi} \mathbf{w}_n + \mathbf{b}$. Because this state is reachable from a valid initial state with one extra molecule of A_1 , we must be able to produce a YES voter from it. By assumption of a speed fault free CRD, this must be a fast path: for all f , there is an n such that $\mathbf{w}_n + \mathbf{b} \xRightarrow{f} \mathbf{z}_n$ by an f -fast path q_n , and \mathbf{z}_n contains a YES voter.

In the remainder of the proof, we will first argue that q_n cannot be Ω -friendly for any n . Finally, we will argue that there is a bound f_{\max} such that for any n , q_n cannot be f_{\max} -fast, contradicting the speed fault free assumption.

Suppose q_n is Ω -friendly. Then by Lemma 4.9, there are $\mathbf{o}, \mathbf{o}', \mathbf{o}'' \in \mathbb{N}^{\Omega}$, $\mathbf{z}'_n, \mathbf{z}''_n \in \mathbb{N}^{\Lambda}$ such that $\mathbf{o} + \mathbf{w}_n \xRightarrow{\Omega} \mathbf{o}' + \mathbf{z}'_n$ and $\mathbf{o} + \mathbf{b} \xRightarrow{\Omega} \mathbf{o}'' + \mathbf{z}''_n$ and $\mathbf{z}'_n + \mathbf{z}''_n = \mathbf{z}_n$. Since \mathbf{z}_n contains a YES voter, \mathbf{z}'_n or \mathbf{z}''_n must contain a YES voter. This is a contradiction since \mathbf{b} and \mathbf{w}_n were chosen above such that they remain committed even if any amount of Ω is added to them.

Thus any q_n must begin with a (possibly empty) Ω -friendly portion, followed by a reaction α_n that is not Ω -friendly. Let \mathbf{u}_n be the state immediately before this reaction occurs in each q_n . If for all f , there is a q_n that is f -fast, it must be that \mathbf{u}_n contains count f of some species X_n that is not in Ω (otherwise, α_n would be Ω -friendly). Consider $f > 2c$ where c is the constant from Lemma 4.12. Since the initial portion of q_n that leads to \mathbf{u}_n is Ω -friendly, we have $\mathbf{w}_n + \mathbf{b} \xRightarrow{\Omega} \mathbf{u}_n$ and Lemma 4.9 applies. Consequently, $\exists \mathbf{o}, \mathbf{o}', \mathbf{o}'' \in \mathbb{N}^\Omega$ and $\mathbf{u}', \mathbf{u}'' \in \mathbb{N}^\Lambda$ such that $\mathbf{o} + \mathbf{w}_n \xRightarrow{\Omega} \mathbf{o}' + \mathbf{u}'_n$ and $\mathbf{o} + \mathbf{b} \xRightarrow{\Omega} \mathbf{o}'' + \mathbf{u}''_n$ and $\mathbf{u}'_n + \mathbf{u}''_n = \mathbf{u}_n$. Thus either \mathbf{u}'_n or \mathbf{u}''_n must contain at least $f/2$ of X_n . Since $\mathbf{w}_{n'}$ are nondecreasing and are larger than n' on Ω , for large enough n' , $\mathbf{w}_{n'}$ from W_e exceeds $\mathbf{o} + \mathbf{w}_n$ and $\mathbf{w}_{n'}$ from W_0 exceeds $\mathbf{o} + \mathbf{b}$. But then we obtain a contradiction of condition (4) in Lemma 4.12. \square

Acknowledgements. We thank Damien Woods, Anne Condon, Chris Thachuk, Bonnie Kirkpatrick, Monir Hajiaghayi, and Ján Maňuch for useful discussions.

References

- [1] Dana Angluin, James Aspnes, Zoë Diamadi, Michael Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Computing*, 18:235–253, 2006. Preliminary version appeared in PODC 2004.
- [2] Dana Angluin, James Aspnes, and David Eisenstat. Stably computable predicates are semi-linear. In *PODC 2006: Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pages 292–299, New York, NY, USA, 2006. ACM Press.
- [3] Dana Angluin, James Aspnes, and David Eisenstat. Fast computation by population protocols with a leader. *Distributed Computing*, 21(3):183–199, September 2008. Preliminary version appeared in DISC 2006.
- [4] Luca Cardelli. Strand algebras for DNA computing. *Natural Computing*, 10(1):407–428, 2011.
- [5] Luca Cardelli and Attila Csikász-Nagy. The cell cycle switch computes approximate majority. *Scientific Reports*, 2, 2012.
- [6] E. Cardoza, Richard J. Lipton, and Albert R. Meyer. Exponential space complete problems for Petri nets and commutative semigroups (preliminary report). In *STOC 1976: Proceedings of the 8th annual ACM Symposium on Theory of Computing*, pages 50–54. ACM, 1976.
- [7] Ho-Lin Chen, David Doty, and David Soloveichik. Deterministic function computation with chemical reaction networks. *Natural Computing*, 2013. to appear. Preliminary version appeared in DNA 2012.
- [8] Yuan-Jyue Chen, Neil Dalchau, Niranjan Srinivas, Andrew Phillips, Luca Cardelli, David Soloveichik, and Georg Seelig. Programmable chemical controllers made from DNA. *Nature Nanotechnology*, 8(10):755–762, 2013.
- [9] Anne Condon, Alan Hu, Ján Maňuch, and Chris Thachuk. Less haste, less waste: On recycling and its limits in strand displacement systems. *Journal of the Royal Society Interface*, 2:512–521, 2012. Preliminary version appeared in DNA 2011.
- [10] Leonard E. Dickson. Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors. *American Journal of Mathematics*, 35(4):413–422, October 1913.

- [11] David Doty. Timing in chemical reaction networks. In *SODA 2014: Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 772–784, January 2014.
- [12] Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*, 81(25):2340–2361, 1977.
- [13] Richard M Karp and Raymond E Miller. Parallel program schemata. *Journal of Computer and system Sciences*, 3(2):147–195, 1969.
- [14] Jérôme Leroux. Vector addition systems reachability problem (a simpler solution). In *The Alan Turing Centenary Conference*, pages 214–228, June 2012.
- [15] Carl A Petri. Communication with automata. Technical report, DTIC Document, 1966.
- [16] David Soloveichik, Matthew Cook, Erik Winfree, and Jehoshua Bruck. Computation with finite stochastic chemical reaction networks. *Natural Computing*, 7(4):615–633, 2008.
- [17] David Soloveichik, Georg Seelig, and Erik Winfree. DNA as a universal substrate for chemical kinetics. *Proceedings of the National Academy of Sciences*, 107(12):5393, 2010. Preliminary version appeared in DNA 2008.