

# Reachability bounds for chemical reaction networks and strand displacement systems

Anne Condon · Bonnie Kirkpatrick ·  
Ján Maňuch

Published online: 22 December 2013

© The Author(s) 2013. This article is published with open access at Springerlink.com

**Abstract** Chemical reaction networks (CRNs) and DNA strand displacement systems (DSDs) are widely-studied and useful models of molecular programming. However, in order for some DSDs in the literature to behave in an expected manner, the initial number of copies of some reagents is required to be fixed. In this paper we show that, when multiple copies of all initial molecules are present, general types of CRNs and DSDs fail to work correctly if the length of the shortest sequence of reactions needed to produce any given molecule exceeds a threshold that grows polynomially with attributes of the system.

**Keywords** Chemical reaction networks · Strand displacement systems · Reachability bounds

## 1 Introduction

DNA strand displacement systems (DSDs) (Yurke and Mills 2003; Zhang et al. 2007) and chemical reaction networks (CRNs) (Cook et al. 2009; Soloveichik 2009, 2008) are important molecular programming models. DSDs provide sophisticated molecular realizations of logic circuits and even artificial neurons (Qian and Winfree 2011; Qian et al. 2011b), while CRNs elegantly express chemical programs that can then be translated into DSDs (Chen et al. 2012; Soloveichik et al. 2008, 2010). CRNs

and thus DSDs can in principle simulate Turing-general models of computation (Qian et al. 2011a; Seelig et al. 2006), and DSDs can be energy efficient (Seelig et al. 2006; Soloveichik et al. 2010; Yurke et al. 2000; Zhang and Seelig 2011). It is also possible in principle to recycle molecules in DSDs by running reversible reactions or displacements in both forwards and reverse directions, so that  $t$  steps of the system use just  $O(\log t)$  molecules (Condon et al. 2012; Thachuk and Condon 2012).

However, correct behavior of some published DSDs (Condon et al. 2012; Qian et al. 2011a) requires that an exact numbers of some reactants are present initially, and it is currently impractical to obtain the exact numbers in a wet lab. We previously considered the conditions for a class of CRNs to work correctly when multiple copies of all initial molecules are present and showed that the length of the shortest trace (sequence of reactions) needed to “reach”, i.e., produce, any given molecule is bounded by a polynomial function of some attributes of a CRN in this class (Condon et al. 2012). This reachability upper bound reveals important limits of molecular programs that fall in the class covered by our result: we cannot write such programs that run correctly in a closed chemical system and for which the number of steps (reactions) of the program is sufficiently large relative to the volume of initial reagents.<sup>1</sup>

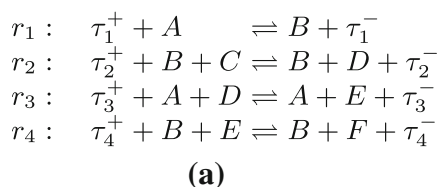
In this work we provide two new reachability upper bounds that significantly extend our earlier work. The first new theorem applies to *tagged* CRNs which, as we explain below, are important because they can be translated into DSDs of comparable volume that can simulate the CRN

---

A. Condon · B. Kirkpatrick · J. Maňuch (✉)  
Department of Computer Science, University of British  
Columbia, Vancouver, BC, Canada  
e-mail: jmanuch@cs.ubc.ca

J. Maňuch  
Department of Mathematics, Simon Fraser University, Burnaby,  
BC, Canada

<sup>1</sup> Volume refers to the physical volume of all the molecules. The initial volume can be approximated by the total number of all types of reagents in the initial configurations.



Reaction	Signals	Tags
	$A, C$	$\tau_1^+, \tau_2^+, \tau_3^+, \tau_4^+$
$+r_1$	$B, C$	$\tau_1^-, \tau_2^+, \tau_3^+, \tau_4^+$
$+r_2$	$B, D$	$\tau_1^-, \tau_2^-, \tau_3^+, \tau_4^+$
$-r_1$	$A, D$	$\tau_1^+, \tau_2^-, \tau_3^+, \tau_4^+$
$+r_3$	$A, E$	$\tau_1^+, \tau_2^-, \tau_3^-, \tau_4^+$
$+r_1$	$B, E$	$\tau_1^-, \tau_2^-, \tau_3^-, \tau_4^+$
$+r_4$	$B, F$	$\tau_1^-, \tau_2^-, \tau_3^-, \tau_4^-$

(b)

**Fig. 1** Example of a simple tagged chemical reaction network (CRN). **a** List of reactions, all of which are reversible. **b** Changes in signal and tag species as reactions occur. The first row lists species present initially. The left column of subsequent rows lists the reaction

applied, with *plus* indicating the forward and *minus* indicating the backwards direction. Two right columns show the signal and tag species, respectively, after the reaction has been applied

traces. The second new theorem applies to a broader class of DSDs than does the translated version of our first result. In the rest of this introduction we motivate our results in more detail. Sections 2 and 3 provide technical details of both theorems. We list some open questions in Sect. 4.

### 1.1 New result for chemical reaction networks (CRNs)

Figure 1a illustrates a CRN of the type to which our new result applies (a formal definition of CRN is in Sect. 2). Each reaction  $r_i$  is reversible, and has unique *tag* species  $\tau_i^+$  and  $\tau_i^-$  on its left and right sides respectively. We explain later why we focus on tagged CRNs, and also explain why we ignore reaction rate constants in our example and results.

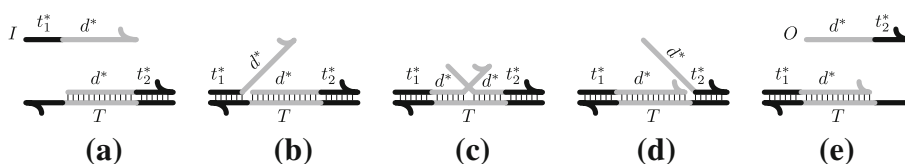
When a single copy of each species in the set  $\{A, C, \tau_1^+, \tau_2^+, \tau_3^+, \tau_4^+\}$  is initially present, it takes six reaction steps to produce the product  $F$ , and to do so, reaction  $r_1$  must run in the forwards direction, then later run backwards, then forwards again, cf. Fig. 1b. However, if another copy of  $A$  is present initially then  $F$  can be generated with just four reactions. The behavior of the system with two copies does not mirror its behavior with one copy; in this sense it is incorrect. While for this simple example it might not seem important how many steps are needed to produce a particular product, it is critically important in contexts where the product is the result of a computation and an erroneous result could be produced as a result of cross-talk, or short-circuiting of multiple copies of the intended computation.

In this paper, our notion of correctness is that of *copy tolerance* (Condon et al. 2012). We say that a CRN  $\mathbf{C}$  is  $x$ -copy-tolerant if the length of the shortest trace that produces any species  $s$  in  $\mathbf{C}$  and in  $\mathbf{C}^{(x)}$  is the same, where  $\mathbf{C}^{(x)}$  is the CRN with the same reactions as  $\mathbf{C}$  but with  $x$  initial copies of each initial molecule of  $\mathbf{C}$ . A system is copy-tolerant if it is  $x$ -copy-tolerant for all  $x$ . The CRN of Fig. 1

is not 2-copy-tolerant. Copy-tolerance is a weak notion of correctness; if a CRN  $\mathbf{C}$  is not 2-copy tolerant then, for example,  $\mathbf{C}$  also fails to satisfy the stronger requirement that each possible trace of  $\mathbf{C}$  in the 2-copy setting is an interleaving of two possible traces in the single copy setting. We chose to work with a weak notion of correctness because it makes our results stronger, i.e., they apply also to notions of correctness that are stronger than copy-tolerance.

Our first reachability upper bound, Theorem 2, shows that in order for a tagged CRN  $\mathbf{C}$  to be copy-tolerant, the number of steps needed for  $\mathbf{C}$  to produce any given species must be suitably bounded. The bound is a polynomial function of the volume and other attributes of  $\mathbf{C}$ .

We prove our result for *tagged* CRNs—CRNs with a unique species on the left and right side of each reaction (Fig. 1)—for two reasons. First, the tags make it possible for us to prove strong results. The second reason stems from the fact that our ultimate goal is to prove limits on the power of DSDs, which can be realized with DNA strands, rather than for CRNs which are a useful theoretical abstraction. When translating an “untagged” CRN to a DSD, two sets of auxiliary DNA strand complexes, which we refer to as transformers, are introduced per reaction of the CRN, one set for each side of the reaction. Each set of transformers includes unique strands that do not otherwise appear in the DSD. The CRN tag species represent the sets of transformer DNA strands. Put another way, to translate an untagged CRN to a DSD using current methods, it is necessary to first add tags to the CRN and then map the tags to the sets of transformer species. Thus, by proving a reachability upper bound for a tagged CRN, we are obtaining a result for the DSD realization of the corresponding untagged CRN. The result would apply also to other realizations of CRNs, perhaps even using molecules other than DNA, in which transformer molecules are needed in the realization. Our earlier result (Condon et al. 2012) did not apply to general tagged CRNs.



**Fig. 2** Strand displacement. **a** An unbound DNA strand  $I$ , with a short toehold (dark line) and long-domain (lighter line), plus a duplex consisting of a template strand  $T$  and a third strand  $O$  that is bound to  $T$ . **b**  $I$  binds to  $T$  via its toehold. **c** Through a process of branch

migration, the long-domain of  $I$  becomes bound to  $T$ , displacing bonds of  $O$ . **d**  $O$  is bound to  $T$  by only a toehold. **e** The toehold bonds break, making  $O$  unbound

Unlike the example of Fig. 1, chemical reactions have associated kinetic rate constants that, along with species counts, determine reaction propensities (Soloveichik 2009; Soloveichik et al. 2008). In particular, a CRN behaves stochastically if multiple reactions are applicable to the molecules available at one or more points in the sequence of reactions. However, in examples such as the stack machine of Qian et al. (2011a) and the Gray code counter of Condon et al. (2012), correctness of the CRN does not depend on the relative propensities of applicable reactions (although the expected time to complete the simulation of the CRN does depend on those propensities). Since our results are expressed in terms of number of reactions rather than reaction propensities, they apply to stochastic CRNs. We can interpret our reachability result as a hitting time in the stochastic context where a hitting time is the minimum number of reactions required to reach a goal state from the initial state.

### 1.2 New result for strand displacement systems (DSDs)

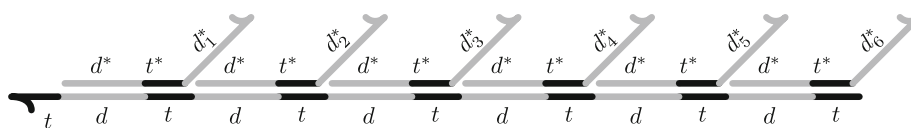
The second main contribution of this paper is a limit on the types of DSDs that are correct in multi-copy settings. In (toehold-mediated) strand displacement (Fig. 2), an initially unbound “signal” strand  $I$  binds to a “template”  $T$ , causing another signal strand  $O$  that was initially bound to  $T$  to become unbound. DSDs are collections of strands that can change configurations via successive strand displacements in a pre-programmed fashion (Cardelli 2010; Zhang and Seelig 2011; Zhang et al. 2007) we provide a formal definition later. We do not allow other types of strand displacements, such as cooperative strand displacements, where two signal strands are needed to displace one signal strand. We thus refer to the DSDs in this paper as Uncooperative DSDs (UDSDs).

Our first result on tagged CRNs implies a reachability upper bound for DSD realizations of CRNs, but says nothing about DSDs more generally. In Theorem 8 we elucidate this simple upper bound which is obtained by applying the CRN result to limited types of DSDs, those whose signal strands consist of exactly two domains: a toehold and a long-domain. However, since the signal types are limited, this result does not apply to general DSDs. This is because, while tagged CRNs can be translated to DSDs having parameters such as

the volume and the number of types of reactants polynomial in the volume of the CRN (Soloveichik et al. 2011), it is not clear whether the converse is true. To see why, consider signal strands that have three domains: a toehold and two long-domains such that they each start with the same long-domain  $d^*$  and toehold  $t^*$ , and end with a distinct long-domain. Assume there are  $\delta$  different types of these signal strands where  $\delta$  is the number of long-domains on the template we will consider. Note for the DSD template having  $\delta$  long-domains, over the course of several displacements, there are factorially many different configurations—ways in which signal strands are bound to the template. Figure 3 provides a simple example where any permutation of the signal species could bind to the template. Now, we want to create a tagged CRN that is equivalent to this DSD. Such a tagged CRN in which each template configuration is a distinct species would thus have the number of distinct species and reactions factorial in the volume (number of toeholds and long-domains) of the DSD. Since each reaction in the tagged CRN requires a unique tag which needs to be present in the initial configuration, the overall volume of the tagged CRN would be also factorial in the volume of the DSD. It is not clear how else to translate such a DSD to a (tagged) CRN of comparable volume.

Can “long” computations be correctly performed by DSDs, even in the presence of many copies? Our second reachability upper bounds for UDSDs, Theorems 9 and 10, answer this in the negative, showing that, if sufficiently many copies are present, then any unbound DNA strand that can be produced (i.e., reached) by a sequence of strand displacements can always be reached within a number of displacements that grows at most polynomially in the volume of the single-copy UDSD. Thus, for example, we cannot write DSD programs that run correctly in the multi-copy setting and for which the minimum number of displacements needed to produce some given signal strand is exponential in the initial volume.

As further motivation, we describe another application of our DSD reachability bound. The CRN of Fig. 4 describes a traditional 3-bit binary counter. Initially, three species, namely  $0_3$ ,  $0_2$  and  $0_1$  represent the bits 0 at each index of the counter. Exactly one reaction can advance the



**Fig. 3** A template indicated by the *bottom line* has 6 long-domains and  $6! = 720$  possible configurations. The signals are the bent *top lines*. *Dark lines* are toeholds and the *lighter ones* are long-domains. The template contains  $\delta = 6$  toehold-long-domain-toehold blocks. In each block, any one of the signal species shown may be bound. Thus

$$\begin{aligned} (1) \mathbf{0}_1 & \quad \quad \quad \Rightarrow \mathbf{1}_1 \\ (2) \mathbf{0}_2 + \mathbf{1}_1 & \quad \Rightarrow \mathbf{1}_2 + \mathbf{0}_1 \\ (3) \mathbf{0}_3 + \mathbf{1}_2 + \mathbf{1}_1 & \Rightarrow \mathbf{1}_3 + \mathbf{0}_2 + \mathbf{0}_1 \end{aligned}$$

**Fig. 4** Binary counter CRN

counter from each value (all in the forward direction), until the counter reaches  $\mathbf{1}_3 \mathbf{1}_2 \mathbf{1}_1$ . For the  $n$ -bit generalization of this counter, the number of species is just  $2n$  (two species per bit) while the number of steps is  $2^n$ . Thus the volume is logarithmic in the number of steps. Another very nice feature of this CRN is that it works correctly even if multiple copies of the initial species are present, not only in the sense of being copy-tolerant but also in the sense that the trace of the multi-copy system is an interleaving of traces of the single-copy system, even in the presence of cross-talk. This follows from the fact that for every  $i$ , to produce each copy of molecule  $\mathbf{1}_i$ , reactions (i) and (i - 1) have to be executed at least once in forward direction, (i - 2) at least twice, ..., (1) at least  $2^{i-2}$  times, which can be proved by induction.

However, if tags are added to the counter in order that it can be translated to a DSD using tags as discussed previously, the volume of species for the DSD realization of the counter becomes exponential in  $n$ . This is because reaction (1) is executed in the forward direction  $2^{n-1}$  times and is never executed in the reverse direction; thus  $2^{n-1}$  copies of the tag on the left side of reaction (1) must be present initially. Is there an alternative (tag-less) DSD realization of the  $n$ -bit CRN binary counter whose volume grows polynomially in  $n$ ? Our DSD result implies that there is no such realization. If there were, then our reachability upper bound implies that in the multi-copy setting the bit  $\mathbf{1}_n$  could be produced in a polynomial number of steps. But since we know that it takes  $2^{n-1}$  steps to produce  $\mathbf{1}_n$  even in the multi-copy setting, we have a contradiction.

## 2 Reachability upper bound for CRNs

In this section we first provide formal definitions of tagged CRNs. We then provide our main technical result, restate this result to obtain our reachability upper bound theorem

the number of possible configurations of this template is  $\delta! = 6!$ . If 6 copies of the signal  $t^*d^*$  are present as well, they can displace 6 signals shown, which can subsequently displace six  $t^*d^*$  signals resulting in any of the  $6!$  configurations

for copy-tolerant CRNs, compare the bounds of our main theorem of Sect. 2 with our previous result (Condon et al. 2012), and then provide several additional results.

### 2.1 Definition of tagged CRNs

**Notation.** If  $\mathcal{S}$  is a multiset, we will denote the set of distinct elements in  $\mathcal{S}$  as  $\llbracket \mathcal{S} \rrbracket$ . If  $s$  is an element and  $k$  is a positive integer, then  $k \cdot s$  denotes  $k$  copies of  $s$ . For example, a multiset containing three copies of  $a$  and five copies of  $b$ , can be represented as  $\{3 \cdot a, 5 \cdot b\}$ . If  $\mathcal{S}$  is a set and  $k$  is a positive integer, then  $k \cdot \mathcal{S}$  denotes the multiset containing  $k$  copies of each element in  $\mathcal{S}$ . Similarly, if  $\mathcal{S}$  is a multiset, then  $k \cdot \mathcal{S}$  denotes the union of  $k$  copies of  $\mathcal{S}$ . The set operations on multisets are defined in a usual way. Let  $\#\{x \in \mathcal{S}\}$  denote the number of copies of  $x$  in  $\mathcal{S}$ . In addition, we define the intersection  $\mathcal{S} \cap T$  of a multiset  $\mathcal{S}$  and a set  $T$  as  $\mathcal{S} \cap (|\mathcal{S}| \cdot T)$ , i.e.,  $\mathcal{S} \cap T$  contains only elements in  $\llbracket \mathcal{S} \rrbracket \cap T$ , and for each  $x \in \llbracket \mathcal{S} \rrbracket \cap T$ ,  $\#\{x \in \mathcal{S} \cap T\} = \#\{x \in \mathcal{S}\}$ .

**Definition 1 (Tagged CRN)** A tagged chemical reaction network is a tuple  $\mathbf{C} = \langle \mathcal{S}, T, R, \mathcal{S}_0, T_0 \rangle$  with variables defined as follows:

- $\mathcal{S}$  is a set of signal species and  $T$  is the set of tag species, and  $\mathcal{S} \cap T = \emptyset$ .
- $R$  is a set of reversible or irreversible reactions, where each  $r \in R$  is an ordered pair  $(\mathcal{I}_r, \mathcal{P}_r)$  of multisets of signal and tag molecules such that  $\mathcal{I}_r \cap T = \{\tau_r^+\}$  and  $\mathcal{P}_r \cap T = \{\tau_r^-\}$ . Note that each side of each reaction contains exactly one tag molecule and this tag molecule is unique for that reaction. Intuitively, a reaction  $r = (\mathcal{I}_r, \mathcal{P}_r)$  either consumes the molecules in  $\mathcal{I}_r$  and produces the molecules  $\mathcal{P}_r$ , or, if the reaction is reversible, it can also consume  $\mathcal{P}_r$  and produce  $\mathcal{I}_r$ . In the first case, we say that the reaction was applied in the *forward* direction and denote it as  $+r$ , in the second case in the *backward* direction and denote it as  $-r$ . The symbols  $+r$  and  $-r$  will be called *oriented* reactions and we define  $|+r| = |-r| = r$ . We will refer to  $\mathcal{I}_r$  and  $\mathcal{P}_r$  as the *left side* and the *right side* of a forward

reaction  $+r$ , and as the *right side* and the *left side* of a backward reaction  $-r$ .

- $S_0$  is a multiset of signal molecules and  $T_0$  is a multiset of tag molecules present initially at time-step zero. The *volume* of CRN  $C$  is the number of molecules in  $S_0 \cup T_0$ .

Tags limit the number of times a reaction can be applied in the same direction without being applied in the reverse direction. For example, if  $r$  is a reversible reaction and  $T_0$  contains only one copy of  $\tau_r^+$  and no copies of  $\tau_r^-$ , then in any valid trace, the oriented occurrences of  $r$  have to alternate, starting with  $+r$ . If  $r$  is an irreversible reaction and  $T_0$  contains  $x$  copies of  $\tau_r^+$ , then in any valid trace, there are at most  $x$  occurrences of  $+r$  (and no occurrences of  $-r$ ). Limiting the number of tags forces a system to recycle molecules in long traces.

In the following series of definitions, consider a tagged CRN system  $C = \langle S, T, R, S_0, T_0 \rangle$ .

**Definition 2 (Bandwidths)** Define the bandwidth of signal species  $s$  as the maximum number of occurrences of  $s$  in  $\mathcal{I}_r$  or  $\mathcal{P}_r$ , i.e.,  $\max_{r \in R} \{ \#\{s \in \mathcal{I}_r\}, \#\{s \in \mathcal{P}_r\} \}$ . Define the maximum bandwidth  $b_C$  (respectively, *total bandwidth*  $B_C$ ) of  $C$  as the maximum (respectively, the sum) of bandwidth over all signal species in  $S$ . Similarly, the proper bandwidth of signal species  $s$ , the maximum proper bandwidth  $\tilde{b}_C$  and the total proper bandwidth  $\tilde{B}_C$  are defined analogously but using  $\mathcal{I}_r \setminus \mathcal{P}_r$  instead of  $\mathcal{I}_r$  and  $\mathcal{P}_r \setminus \mathcal{I}_r$  instead of  $\mathcal{P}_r$ .

To illustrate the above definition, consider the CRN  $C$  that consists of two reactions,  $A + B \rightleftharpoons A + C$  and  $B + B \rightleftharpoons C$ . Now the respective bandwidths of the species  $A, B$ , and  $C$  are 1, 2, and 1, the maximum bandwidth  $b_C = 2$  and the total bandwidth  $B_C = 4$ . Similarly, the respective proper bandwidths of the species  $A, B$ , and  $C$  are 0, 2, and 1, the maximum proper bandwidth  $\tilde{b}_C = 2$  and the total proper bandwidth  $\tilde{B}_C = 3$ .

**Definition 3 (Numbers of occurrences of tags)** For any reversible reaction  $r \in R$ , let  $t_r$  be the maximum of the number of occurrences of  $\tau_r^+$  or  $\tau_r^-$  in  $T_0$ , i.e.,  $\max\{\#\{\tau_r^+ \in T_0\}, \#\{\tau_r^- \in T_0\}\}$ ; and for any irreversible reaction  $r \in R$ , let  $t_r$  be the number of occurrences of  $\tau_r^+$  in  $T_0$ . Let  $T_C$  be the sum of  $t_r$ 's over all reactions  $r \in R$ .

**Definition 4 ( $x$ -copy CRN)** We define the  $x$ -copy of  $C$ , for  $x \in \mathbb{Z}^+$ , as the CRN  $\langle S, T, R, x \cdot S_0, x \cdot T_0 \rangle$ .

**Definition 5 (Trace)** Let  $\rho = r_1, r_2, \dots, r_m$  be a sequence of oriented reactions where  $|r_i| \in R$  for all  $i$ . For oriented reaction  $r$  if  $\text{sign}(r) = +$ , let  $\mathcal{A}_r = \mathcal{I}_r$  and  $\mathcal{B}_r = \mathcal{P}_r$  whereas if  $\text{sign}(r) = -$ , let  $\mathcal{A}_r = \mathcal{P}_r$  and  $\mathcal{B}_r = \mathcal{I}_r$ . The configuration of the system at each step  $i$  is defined as

$(S_i, T_i)$  where  $S_i = (S_{i-1} \setminus (\mathcal{A}_{r_i} \cap S)) \cup (\mathcal{B}_{r_i} \cap S)$  and, similarly,  $T_i = (T_{i-1} \setminus (\mathcal{A}_{r_i} \cap T)) \cup (\mathcal{B}_{r_i} \cap T)$ . A reaction sequence  $\rho$  is *valid* if  $\mathcal{A}_{r_i} \cap S \subseteq S_{i-1}$  and  $\mathcal{A}_{r_i} \cap T \subseteq T_{i-1}$  for all  $i$ , meaning that for each molecule in  $\mathcal{A}_{r_i}$  there must be one in  $S_{i-1} \cup T_{i-1}$ . A trace is a valid reaction sequence.

### 2.2 The main upper bound

Our main upper bound, Theorem 1, shows that in the multi-copy setting, any product of a tagged CRN can be produced within a number of reactions that is bounded by a function of the number of signal species, the bandwidth, and the number of tags of the CRN.

**Theorem 1** *Let  $C = \langle S, T, R, S_0, T_0 \rangle$  be a tagged CRN and let  $s_{\text{end}} \in S$ . If some trace of  $C$  produces  $s_{\text{end}}$ , then in a  $(|S| - \|\mathcal{S}_0\| + 1)(b_C + \tilde{b}_C T_C / 2) \leq |S| b_C (T_C / 2 + 1)$ -copy CRN of  $C$ , the length of the shortest trace that produces  $s_{\text{end}}$  is at most  $(|S| - \|\mathcal{S}_0\|)(b_C + \tilde{b}_C T_C / 2) T_C \leq (|S| - 1) b_C (T_C / 2 + 1) T_C$ .*

*Proof* Let  $\rho = r_1, r_2, \dots, r_m$  be a valid sequence of oriented reactions in a single-copy system producing  $s_{\text{end}}$  starting from the initial set  $S_0 \cup T_0$ . We will construct a reaction sequence  $\rho'$  that also produces  $s_{\text{end}}$  in a multi-copy CRN and satisfies the length-bound of the theorem, by first constructing “unidirectional” shortened reaction subsequences by eliminating all forward-backward pairs of reactions  $\{+r, -r\}$  in subsequences of  $\rho$ , and then showing that in a multiple-copy setting the intermediate signals required to drive the synthesis can instead be produced by repeating these shortened reaction subsequences of  $\rho$ . Throughout the proof we will illustrate the construction on the CRN and the corresponding trace from Fig. 1.

Consider any prefix of this sequence, say  $\rho_i = r_1, \dots, r_i$ . Construct a new sequence  $\rho'_i$  by randomly pairing  $+r$  with  $-r$ , for any reaction  $r \in R$ , and removing these pairs from the sequence, until no such pairs can be formed, i.e.,  $\rho'_i$  does not contain either  $+r$  or  $-r$ , for every  $r \in R$ . For example, consider  $\rho'_6 = +r_1, +r_2, -r_1, +r_3, +r_1, +r_4$  from Fig. 1b. Then  $\rho'_6 = +r_2, +r_3, +r_1, +r_4$  or  $\rho'_6 = +r_1, +r_2, +r_3, +r_4$ , depending on the choice of the  $+r_1$  and  $-r_1$  pair.

The constructed reaction sequence  $\rho'_i$  has the same effect on the final number of signals as  $\rho_i$ . However,  $\rho'_i$  might not be a valid reaction sequence starting at the same initial configuration as  $\rho_i$  since some reactants might be missing when running a reaction in  $\rho'_i$ . To avoid that we will start with a sufficient number of copies of signals in  $S_0 \cup T_0$  and run each  $\rho'_i$  that produces a new signal  $s_j$  sufficient number of times so that we have sufficient number of copies of  $s_j$  for all remaining executions of such shortened reaction subsequences. For example, in  $\rho'_6 = +r_2$ ,

$+r_3, +r_1, +r_4$  the missing first reaction  $+r_1$  produces a signal  $B$  which is used by the subsequent reaction  $+r_2$ . We can provide the missing signal  $B$  by running a shortened sequence  $\rho'_1$  that produces  $B$  before executing sequence  $\rho'_6$ . In what follows we will argue that if we start in a configuration with a sufficient number of copies of signals in  $\mathcal{S}_0 \cup \mathcal{T}_0 \cup S_{i-1}$ , the constructed reaction sequence  $\rho'_i$  becomes valid.

Let  $S'$  be the set of signal molecules appearing on the left hand side of reactions in  $\rho'_i$ . Now, let us see what happens if we apply this sequence on the initial set  $\mathcal{S}_0 \cup \mathcal{T}_0 \cup k \cdot S'$ , where  $k$  is sufficiently large so that the reaction sequence is valid. We can make the following observations:

- (1) The final number of copies of each signal species is the same as if we would apply  $\rho_i$  on  $\mathcal{S}_0 \cup \mathcal{T}_0 \cup k \cdot S'$ . Hence, the final configuration contains  $k \cdot S'$ .
- (2) For each reaction  $r \in R$ ,  $\rho'_i$  contains either only forward or only backward occurrences of  $r$  (or no occurrences), and their number is limited by the number  $t_r$  of corresponding tags in  $\mathcal{T}_0$ . As a consequence, the length of  $\rho'_i$  is at most  $T_C$ .
- (3) Consider a signal molecule  $s \in S'$ . Each reaction in  $\rho'_i$  removes or adds at most  $b_C$  copies of  $s$  and the length  $\ell$  of  $\rho'_i$  is at most  $T_C$ . We will show that before each reaction in  $\rho'_i$ , there are at least  $k - \tilde{b}_C T_C / 2$  copies of  $s$ . Assume that after the first  $j$  reactions, the number of copies of  $s$  is less than  $k - \tilde{b}_C \ell / 2$ . If  $j \leq \ell / 2$ , then the first  $j$  reactions of  $\rho'_i$  could remove at most  $\tilde{b}_C j \leq \tilde{b}_C \ell / 2$  copies of  $s$ , and there were at least  $k$  copies present initially, a contradiction. If  $j > \ell / 2$ , then there are less than  $\ell / 2$  reactions left, and each of them adds at most  $\tilde{b}_C$  copies of  $s$ . Since by (1), the final number of copies of  $s$  is at least  $k$ , we have a contradiction again. Hence, the number of copies of  $s$  before any reaction of  $\rho'_i$  is at least  $k - \tilde{b}_C \ell / 2 \geq k - \tilde{b}_C T / 2$ .
- (4) Hence, it follows that if we set  $k = b_C + \tilde{b}_C T_C / 2$ , then before each reaction in  $\rho'_i$ , there are at least  $b_C$  copies of any signal in  $S'$ , and hence, the reaction sequence is valid. Note that this is true even if we randomly permute reactions in  $\rho'_i$ .

For each signal  $s$  appearing in the single-copy trace and not appearing in the initial set  $\mathcal{S}_0$ , let  $r_{\text{index}(s)}$  be the first reaction in  $\rho$  which produces a copy (or more) of  $s$ . Let  $s_1, \dots, s_n$  be the sequence of all signals not in  $\mathcal{S}_0$  ordered by their indices, i.e.,  $\text{index}(s_1) \leq \text{index}(s_2) \leq \dots \leq \text{index}(s_n)$ . In our example from Fig. 1, we have  $\text{index}(B) = 1$ ,  $\text{index}(D) = 2$ ,  $\text{index}(E) = 4$  and  $\text{index}(F) = 6$ . Hence, we order signals as follows:  $s_1 = B$ ,  $s_2 = D$ ,  $s_3 = E$  and  $s_4 = F$ .

Without loss of generality we can assume  $s_n = s_{\text{end}}$ . Let  $S_i = \{s_1, \dots, s_i\}$ . We can make one additional observation:

- (5) For each  $s_i$ , the left side of each reaction in  $\rho'_{\text{index}(s_i)}$  contains only signals in  $\llbracket \mathcal{S}_0 \rrbracket \cup S_{i-1}$ . By (4), if we start in a configuration which contains the multiset of signals and tags  $\mathcal{S}_0 \cup \mathcal{T}_0 \cup (b_C + \tilde{b}_C T_C / 2) \cdot (\llbracket \mathcal{S}_0 \rrbracket \cup S_{i-1})$ ,  $\rho'_{\text{index}(s_i)}$  is a trace producing a copy of  $s_i$ .

### 2.2.1 Construction of reaction sequence

- (S1) Start with the initial set containing  $b_C + \tilde{b}_C T_C / 2$  copies of  $\llbracket \mathcal{S}_0 \rrbracket$  and the empty sequence of reactions.
- (S2) For each  $i = 1, \dots, n$ : add  $b_C + \tilde{b}_C T_C / 2$  copies of  $\mathcal{S}_0 \cup \mathcal{T}_0$  to the initial set and append  $b_C + \tilde{b}_C T_C / 2$  times sequence  $\rho'_{\text{index}(s_i)}$  to the constructed sequence of reactions.

Before we proceed with proving that this construction is producing a valid reaction sequence, let us illustrate it on the CRN from Fig. 1. Since  $T_C = 4$  and  $b_C = \tilde{b}_C = 1$ , we have  $b_C + \tilde{b}_C T_C / 2 = 3$ . The construction starts by putting  $\{3 \cdot A, 3 \cdot C\}$  into the initial set and proceeds in four steps:

1.  $\rho_{\text{index}(B)} = +r_1$ , and hence,  $\rho'_{\text{index}(B)} = +r_1$ . We add  $\{3 \cdot A, 3 \cdot C\} \cup 3 \cdot \mathcal{T}_0$  into the initial set and start constructing a new reaction sequence with  $+r_1, +r_1, +r_1$ .
2.  $\rho_{\text{index}(D)} = +r_1, +r_2$ , and hence,  $\rho'_{\text{index}(D)} = +r_1, +r_2$ . We add  $\{3 \cdot A, 3 \cdot C\} \cup 3 \cdot \mathcal{T}_0$  into the initial set and append  $+r_1, +r_2, +r_1, +r_2, +r_1, +r_2$  to the constructed sequence.
3.  $\rho_{\text{index}(E)} = +r_1, +r_2, -r_1, +r_3$ , and hence,  $\rho'_{\text{index}(E)} = +r_2, +r_3$ . We add  $\{3 \cdot A, 3 \cdot C\} \cup 3 \cdot \mathcal{T}_0$  into the initial set and append  $+r_2, +r_3, +r_2, +r_3, +r_2, +r_3$  to the constructed sequence.
4.  $\rho_{\text{index}(F)} = +r_1, +r_2, -r_1, +r_3, +r_1, +r_4$ , and we choose the second option  $\rho'_{\text{index}(F)} = +r_1, +r_2, +r_3, +r_4$ . We add  $\{3 \cdot A, 3 \cdot C\} \cup 3 \cdot \mathcal{T}_0$  into the initial set and append  $+r_1, +r_2, +r_3, +r_4, +r_1, +r_2, +r_3, +r_4, +r_1, +r_2, +r_3, +r_4$ , to the constructed sequence.

Hence, the construction requires 15 copies of  $\mathcal{S}_0$  and 12 copies of  $\mathcal{T}_0$ , and the constructed reaction sequence contains 27 reactions. This is not the most efficient sequence. As we have seen in the introduction, there is a reaction sequence of length four that uses only two copies of  $\mathcal{S}_0$  and one copy of  $\mathcal{T}_0$ , and produces a signal species  $F$ . However, this general construction guarantees that the required number of copies of the initial set and the length of the sequence is polynomial for any CRN. The configuration of

	$\{3 \cdot A, 3 \cdot C\}$
$3 \cdot \rho'_{\text{index}(B)}$	$\{3 \cdot A, 6 \cdot C, 3 \cdot B\}$
$3 \cdot \rho'_{\text{index}(D)}$	$\{3 \cdot A, 6 \cdot C, 6 \cdot B, 3 \cdot D\}$
$3 \cdot \rho'_{\text{index}(E)}$	$\{6 \cdot A, 6 \cdot C, 6 \cdot B, 3 \cdot D, 3 \cdot E\}$
$3 \cdot \rho'_{\text{index}(F)}$	$\{6 \cdot A, 6 \cdot C, 9 \cdot B, 3 \cdot D, 3 \cdot E, 3 \cdot F\}$

**Fig. 5** The intermediate configurations after each step of the construction applied to the CRN in Fig. 1a. The multisets in the second column show only the signal species

the system after each step is shown in Fig. 5. Note that after each step the configuration contains at least three copies of each species produced so far. We will show that this is always the case in the following claim, which also proves that the constructed sequence is valid.

**Claim 1** *After each step  $i$  in (S2), the constructed sequence is valid and the final configuration contains  $b_C + \tilde{b}_C T_C/2$  copies of each signal in  $\llbracket S_0 \rrbracket \cup S_i$ .*

*Proof* Proof by induction: *Base case:* For  $i = 0$ , after (S1), we have  $b_C + \tilde{b}_C T_C/2$  copies of each signal in  $\llbracket S_0 \rrbracket$  and the empty sequence of reactions is valid. *Induction step:* Inductive assumption: before step  $i$ , we have  $b_C + \tilde{b}_C T_C/2$  copies of each signal in  $\llbracket S_0 \rrbracket \cup S_{i-1}$  and the sequence constructed so far is valid. By (5), if we add a copy of  $S_0 \cup T_0$  and apply the reaction sequence  $\rho'_{\text{index}(s_i)}$  on the current configuration, the trace is valid. By (1), this newly added part (a copy of  $S_0 \cup T_0$  and reactions in  $\rho'_{\text{index}(s_i)}$ ) will not decrease the number of any signal. Finally,  $\rho'_{\text{index}(s_i)}$  must contain the last reaction of  $\rho_{\text{index}(s_i)}$ , i.e.,  $r_{\text{index}(s_i)}$  which produces at least one copy of  $s_i$ . If we repeat this  $b_C + \tilde{b}_C T_C/2$  times, we will still have at least  $b_C + \tilde{b}_C T_C/2$  copies of signals in  $\llbracket S_0 \rrbracket \cup S_{i-1}$  plus  $b_C + \tilde{b}_C T_C/2$  copies of  $s_i$ .  $\square$

The bound: The construction uses  $(n + 1)(b_C + \tilde{b}_C T_C/2)$  copies of  $S_0$ ,  $n(b_C + \tilde{b}_C T_C/2)$  copies of  $T_0$  and repeats  $n(b_C + \tilde{b}_C T_C/2)$  times the trace  $\rho'_{\text{some index}}$ . By (2), the length of each  $\rho'_{\text{some index}}$  trace is at most  $T_C$ , hence the total length of the constructed sequence is at most  $n(b_C + \tilde{b}_C T_C/2)T_C$ . Furthermore,  $n$  can be bounded by  $|S| - |\llbracket S_0 \rrbracket|$ .  $\square$

We remark that this result does not hold for the untagged CRNs, cf. Example 3.4 in our earlier paper (Condon et al. 2012), where the following CRN was presented which requires an exponential number of steps to produce  $s_{\text{end}}$  even in an  $\infty$ -copy of this CRN:

$$s_i + s_i \rightleftharpoons s_{i+1} + s_0, \quad \text{for } i = 0, \dots, k - 1,$$

with the initial set containing  $k$  copies of  $s_0$ . Note that since all reactions are balanced the volume of this system stays

constant. However, since in this CRN in any shortest trace producing  $s_{\text{end}}$ , all reactions are applied in the forward direction, if we would tag this CRN, the trace producing  $s_{\text{end}}$  would require that the initial multiset of tag molecules  $T_0$  contains an exponential number of tags. It is also interesting to observe where the proof of Theorem 1 fails for the untagged CRNs. In observation (2) in the proof we were able to bound the length of the shortened sequence of reactions  $\rho'_i$  by  $T_C$ , which would not be possible in an untagged CRN.

Next, we restate Theorem 1 for copy-tolerant CRNs.

**Theorem 2** *If a tagged CRN  $C = \langle S, T, R, S_0, T_0 \rangle$  is  $|S|b_C(T_C/2 + 1)$ -copy-tolerant and  $s_{\text{end}}$  can be produced in  $C$ , then the length of the shortest trace of  $C$  that produces  $s_{\text{end}}$  is at most  $(|S| - 1)b_C(T_C/2 + 1)T_C$ .*

A natural question is whether we could improve the bound in condition (3) of the proof of Theorem 1 by choosing the “right” permutation of oriented reactions in  $\rho'_i$ . The following example shows that this is not possible in general.

*Example 1* Assume that  $\rho$  contains exactly an even number,  $T$ , of oriented reactions  $+r_1, \dots, +r_T$  designed as follows. First for every partition  $\pi$  of  $\rho$  into two sets  $\rho_1^\pi$  and  $\rho_2^\pi$  of same size, we introduce a new signal  $s_\pi$ . Let  $\Pi$  be the set of all such partitions. Next, we define reactions  $r_1, \dots, r_T$  in such a way that each of these signals is either an input or a product of each reaction:

$$\begin{aligned} \mathcal{I}_{r_i} &= \{s_\pi : r_i \in \rho_1^\pi, \pi \in \Pi\}, \\ \mathcal{P}_{r_i} &= \{s_\pi : r_i \in \rho_2^\pi, \pi \in \Pi\} \end{aligned}$$

We will illustrate this construction for  $T = 4$ , i.e.,  $\rho = +r_1, +r_2, +r_3, +r_4$ . There are six partitions of  $\rho$  into two subsets of size two:

- $\rho_1^\alpha = +r_1, +r_2$  and  $\rho_2^\alpha = +r_3, +r_4$ ,
- $\rho_1^\beta = +r_1, +r_3$  and  $\rho_2^\beta = +r_2, +r_4$ ,
- $\rho_1^\gamma = +r_1, +r_4$  and  $\rho_2^\gamma = +r_2, +r_3$ ,
- $\rho_1^\delta = +r_2, +r_3$  and  $\rho_2^\delta = +r_1, +r_4$ ,
- $\rho_1^\epsilon = +r_2, +r_4$  and  $\rho_2^\epsilon = +r_1, +r_3$ ,
- $\rho_1^\zeta = +r_3, +r_4$  and  $\rho_2^\zeta = +r_1, +r_2$ ,

Hence, the reactions use six signals:  $s_\alpha, s_\beta, s_\gamma, s_\delta, s_\epsilon, s_\zeta$ . Using the definition of inputs and products above, the four constructed reactions are the following:

$$\begin{aligned} r_1 : \quad & s_\alpha + s_\beta + s_\gamma \rightleftharpoons s_\delta + s_\epsilon + s_\zeta \\ r_2 : \quad & s_\alpha + s_\delta + s_\epsilon \rightleftharpoons s_\beta + s_\gamma + s_\zeta \\ r_3 : \quad & s_\beta + s_\delta + s_\zeta \rightleftharpoons s_\alpha + s_\gamma + s_\epsilon \\ r_4 : \quad & s_\gamma + s_\epsilon + s_\zeta \rightleftharpoons s_\alpha + s_\beta + s_\delta \end{aligned}$$

Note that after all reactions in  $\rho$  are applied, the number of copies of any of the signals  $s_\pi$  is not changed, since there

are exactly  $T/2$  reactions in  $\rho$  adding one copy of  $s_\pi$  and  $T/2$  reactions removing one copy of  $s_\pi$ .

Now, we show that for any permutation of the reactions in  $\rho$ , there is a signal molecule with  $k - T/2$  copies when the first  $T/2$  reactions in this order are applied. Since we could easily replace  $\mathcal{I}_{r_i}$  and  $\mathcal{P}_{r_i}$  with  $b \cdot \mathcal{I}_{r_i}$  and  $b \cdot \mathcal{P}_{r_i}$ , the bound  $k - \tilde{b}_C T/2$  in (3) cannot be improved without adding some additional conditions on the CRN. To find the signal molecule with  $k - T/2$  copies after applying the first  $T/2$  reactions, consider the partition  $\pi_0$  of  $\rho$  into the first and the second  $T/2$  reactions of this order. Then the signal  $s_{\pi_0}$  appears in the input set of the first  $T/2$  reactions, and thus, the number of copies of  $s_{\pi_0}$  is  $k - T/2$  after applying the first  $T/2$  reactions.

### 2.2.2 Result for 1-proper tagged CRNs

We next describe a stronger version of our result for a special case. We say that a tagged CRN  $\mathbf{C}$  is  $k$ -proper if each reaction has at most  $k$  reactants which are not catalysts, more formally, for all  $r \in R$ ,  $|\mathcal{I}_r \setminus \{\tau_r^+\} \setminus \mathcal{P}_r| \leq k$  and if  $r$  is reversible, also  $|\mathcal{P}_r \setminus \{\tau_r^-\} \setminus \mathcal{I}_r| \leq k$ .

**Corollary 1** *If there exists a trace in a 1-proper tagged CRN  $\mathbf{C} = \langle S, T, R, S_0, T_0 \rangle$  producing  $s_{\text{end}}$ , then in an  $|\text{Slb}_C$ -copy CRN of  $\mathbf{C}$ , the length of the shortest trace that produces  $s_{\text{end}}$  is at most  $(|S| - 1)b_C T_C$ .*

*Proof* To improve the bound we will strengthen the bound for  $k$  in observations (1–4) in the proof of Theorem 1. In particular, we will show that there is a permutation of reactions in  $\rho'_i$  such that when this permutation of reactions is applied on  $S_0 \cup T_0 \cup b_C \cdot S'$ , the number of copies of any signal species is not below  $b_C - 1$  during any step and the number of copies of any but one signal is not below  $b_C$ . To do this we will borrow the idea from the proof of Theorem 2 in (Condon et al. 2012). Pick the first reaction at random. Since it is a 1-proper reaction, the number of copies of at most one signal species, say  $s$ , is less than  $b_C$ , and if so, by at most one less. By (1), there has to be an unused reaction which would bring this number back to  $b_C$ . We choose this reaction as the second reaction. This brings the number of copies of molecule species  $s$  back to  $b_C$ , but it might decrease the number copies of another species to  $b_C - 1$ . Hence, there is again at most one signal species with fewer than  $b_C$  copies. Repeating this process, we construct the desired permutation of reactions of  $\rho'_i$ .

Using this improved bound of condition (3), we can now modify the construction of the reaction sequence as follows:

(S1) Start with  $b_C$  copies of  $\llbracket S_0 \rrbracket$ .

(S2) For each  $i = 1, \dots, n$ : add  $b_C$  copies of  $S_0 \cup T_0$  and append  $b_C$  times sequence  $\rho'_{\text{index}(s_i)}$ .

The rest of the proof follows analogously to the proof of Theorem 1.  $\square$

### 2.2.3 Comparison with the previous result

In our previous work (Condon et al. 2012), we have showed the following result for untagged CRNs. Untagged CRNs do not put any restriction on how many times reactions are used in forward or backward directions. They can be also thought of as tagged CRNs with an infinite supply of tags [for the exact definition see (Condon et al. 2012)].

**Theorem 6** (Condon et al. 2012). *If there exists a trace in a 1-proper CRN  $\mathbf{C} = \langle S, R, S_0 \rangle$  producing  $s_{\text{end}}$ , then in a  $(B_C + 1)$ -copy CRN of  $\mathbf{C}$ , the length of the shortest trace that produces  $s_{\text{end}}$  is at most  $(B_C + 1)B_C/2 + 1$ .*

Note that  $B_C \leq |\text{Slb}_C|$ . In particular, if the maximum bandwidth is 1, then the number of copies of the system required in both results is  $\Theta(|S|)$  and the number of reactions needed to produce  $s_{\text{end}}$  is bounded by  $O(|S|T_C)$  in our new result and by  $O(|S|^2)$  in the result from (Condon et al. 2012).

**2.2.3.1 The upper bound in the unrestricted case** In the previous subsection, we assumed that a single copy CRN can produce the target signal molecule  $s_{\text{end}}$ . Here we study the case without this assumption. We have the following weaker result:

**Theorem 7** *Consider a CRN  $\mathbf{C} = \langle S, R, S_0 \rangle$  with the maximum bandwidth 1. If  $s_{\text{end}}$  can be produced by an  $\infty$ -copy CRN of  $\mathbf{C}$ , then the length of the shortest trace that produces  $s_{\text{end}}$  is at most  $O(2^{|\text{Rl}|})$  in the  $\infty$ -copy CRN of  $\mathbf{C}$ .*

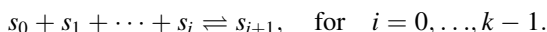
*Proof* Partition  $R$  as follows. In an  $\infty$ -copy CRN, we can assume that we have an unlimited supply of signal molecules in  $S_0$ . Let  $R_1$  be the set of all reactions in  $R$  which can be applied in the initial configuration. Let  $S_1$  be the set of signal molecules in  $S_0$  and those produced by reactions in  $R_1$ . Repeat this procedure until  $S_k$  contains  $s_{\text{end}}$ . Let  $r_i$  be the size of  $R_i$ . We want to estimate how many reaction steps are needed until we can apply the reaction in  $R_k$  that produces  $s_{\text{end}}$ . In the worst case, to apply any reaction in  $R_i$ , we might need signal molecules produced by each reaction in  $R_1 \cup \dots \cup R_{i-1}$ . Let  $b_i$  be an upper bound on the number of reaction steps which will produce all signal molecules required to apply a reaction in  $R_i$ . Hence, we can set  $b_1 = 0$  and  $b_i = \sum_{j=1}^{i-1} r_j(b_j + 1)$ . Note that  $b_{i+1} - b_i = r_i(b_i + 1)$ , and hence,  $b_{i+1} + 1 = r_i(b_i + 1) + b_i + 1 = (b_i + 1)(r_i + 1)$ . And thus,  $b_i = \prod_{j=1}^{i-1} (r_j + 1) - 1$ .



To upper bound  $b_k$  we will argue that the value of  $b_k$  is maximized if  $r_j = 1$  for each  $j = 1, \dots, k$ , and  $k = |R|$ . This is because for any  $n_1, n_2 \geq 1$  such that  $n_1 + n_2 = n$ , it holds that  $n + 1 < (n_1 + 1)(n_2 + 1)$ , i.e., the product could always be increased by replacing the term  $r_j + 1 \geq 3$  in the product with two terms  $r'_j + 1$  and  $r''_j + 1$ , where  $r_j = r'_j + r''_j$ . The claim follows by induction. Therefore, we can upper bound the number of reactions needed to produce  $s_{\text{end}}$  by  $b_k + 1 = \prod_{j=1}^{k-1} (r_j + 1) \leq 2^{|R|-1}$ .  $\square$

The following example shows that the bound in Theorem 7 cannot be improved.

*Example 2* Consider the following CRN with the maximum bandwidth 1:



This CRN contains  $k$  distinct reactions and the number of reaction steps required to produce  $s_k$  is  $2^{k-1}$ , which exactly matches the bound in Theorem 7. Indeed, if we denote the number of reactions needed to produce  $s_i$  by  $n_i$ , then we have  $n_0 = 0$  and  $n_i = n_0 + n_1 + \dots + n_{i-1} + 1$ , and it is easy to check that  $n_i = 2^{i-1}$ , for every  $i \geq 1$ .

### 3 Reachability upper bound for uncooperative DSDs

In this section we first define the type of DSD to which our results apply, along with related notation needed for our results. We then provide our main upper bound, and conclude with a restatement of this result to obtain our reachability upper bound theorem for copy-tolerant DSDs.

#### 3.1 Definition of uncooperative DSDs

In this section we formalize standard features of DSDs as described in the literature and some additional features, so that we can reason rigorously about them in our proofs. Since our model does not allow cooperative strand displacement, we call this model “uncooperative DSD” model (UDSD).

In our definition of UDSDs, we will assume that the basic building blocks are domains where each domain  $d$  has its complementary domain  $d^*$  and  $(d^*)^* = d$ . In practice, domains are built from nucleotide sequences, and it is usually assumed that these are designed in a way so that there are no interactions between domains which are not complementary. In addition, domains are usually divided into two groups, “toeholds” and “long-domains”, based on their lengths (the number of nucleotides). Strands are built by concatenating these basic building blocks. The purpose of toeholds is to initiate branch migration [replacement of one strand (signal) attached to a long

strand (template) by another strand (signal)]. The purpose of long-domains is exactly the opposite: to prevent signal strands from detaching from the template strands without being replaced by another signal strand (as this would require a prohibitive amount of energy). Consistent with existing research, we are working at the domain level of abstraction, and we consider the actual sequence design of domains as future work.

An *Uncooperative DNA strand displacement system (UDSD)* is a pair  $\Delta = (\mathcal{S}, \mathcal{C}_{\text{init}})$  of strands and an initial configuration (secondary structure) for those strands, plus allowable *positional displacements*, defined as follows.

- $\mathcal{S}$  is a finite multiset of *strands*. Strands are composed of subsequences of finite strings of symbols, called *domains*. Domains are partitioned into two groups: *toeholds* and *long-domains*. Corresponding to each domain  $x$  is a complementary domain  $x^*$ ;  $x$  is a toehold if and only if  $x^*$  is.  $\mathcal{S}$  may contain many strands of a given type, where the type of a strand is its sequence of domains. The strands are partitioned into two groups: *signals* and *templates*. A template strand is a sequence of domains beginning and ending with a toehold that alternates between toeholds and long-domains. A signal strand is an arbitrary sequence of domains. There is no bound on the number of toeholds and long-domains of a template or a signal.

We say that the UDSD  $\Delta$  has *simple signals*, if each signal in  $\mathcal{S}$  is composed of exactly one toehold and one long-domain.

- A *configuration* of  $\mathcal{S}$  is a circular graph<sup>2</sup> with the vertex set containing all domains in  $\mathcal{S}$  and the edge set consisting of two types of edges: (i) *adjacency edges* connecting all adjacent domains in the strands of  $\mathcal{S}$  and (ii) *binding edges* connecting some complementary domains, which satisfy the following conditions:
  - (1) Every domain is incident to at most one binding edge. A domain incident to a binding edge, is called *bound*, otherwise, it is called *unbound*.
  - (2) There are no binding edges between domains on template strands.
  - (3) There are no binding edges between domains on signal strands.
  - (4) For each template strand, all domains but one toehold domain are bound. This one unbound toehold is called the *open toehold* of the template.
  - (5) For every signal strand, either all its domains are unbound, in which case we say that the signal strand is *unbound*, or exactly two of its domains

<sup>2</sup> A graph that can be drawn in a way that all vertices lie on a circle and edges lie inside the circle and do not cross. In graph theory, the formal equivalent of circular graphs are outerplanar graphs.

which are adjacent are bound to two adjacent domains on one template strand, in which case we say that the signal strand is *bound*.

In addition, since we assume that a domain is a toehold if and only if the complementary domain is a toehold, and all binding edges are between complementary domains, there are no binding edges between toeholds and long-domains. We will call the connected components of this graph *complexes*. Note that conditions (2), (3) and (5) imply that the configuration is a circular graph, hence we could have omitted it from the above definition. However, we choose to include it so that omitting condition (5) from the above definition yields a valid more general model of DSDs, which we would like to consider in future work.

For example, Fig. 7 shows two configurations, the initial (top) and final (bottom) configurations of a UDSD. This are drawn, not as a typical circular graph, but with domains (vertices) represented by lines (short dark lines represent toeholds and long gray lines represent long-domains), adjacency edges by connected domains and binding edges indicated by complementary domains that are juxtaposed. Most of the signals in these configurations are simple, and there is one complex signal that can bind to two different positions of the first template. The PDs are lined up horizontally with their template positions and given by circles that are numbered according to the order in which they occur.

Let us now provide some intuition behind these conditions. Condition (1) comes from the fact that each nucleotide can form a (hydrogen) bond with only one another nucleotide, and thus the same applies to domains which are sequences of nucleotides. Conditions (2) and (3) are typical assumptions made for the systems which divide strands into templates and signals (as we do). The advantage of such a design is better control of what can and cannot happen in the system. If the UDSD is designed in such a way that no domains in the signals are complementary to each other and similarly, no domains in templates are complementary to each other, then these two conditions are implied and could be dropped from the definition of the configuration. Note that these two conditions imply that the subgraph containing only binding edges is a bipartite graph.

Conditions (4) and (5) are two additional assumptions which we make to prove our results. It is possible that our results hold even if any of these two conditions or both of them are dropped. We leave that as an open problem. Condition (4) guarantees that each configuration is at the minimum free energy. Condition (5) limits how signal strands and template strands interact. If the UDSD is designed in such a way that for each signal and each template there is no scattered substring of the signal of length more than two which is complementary to a

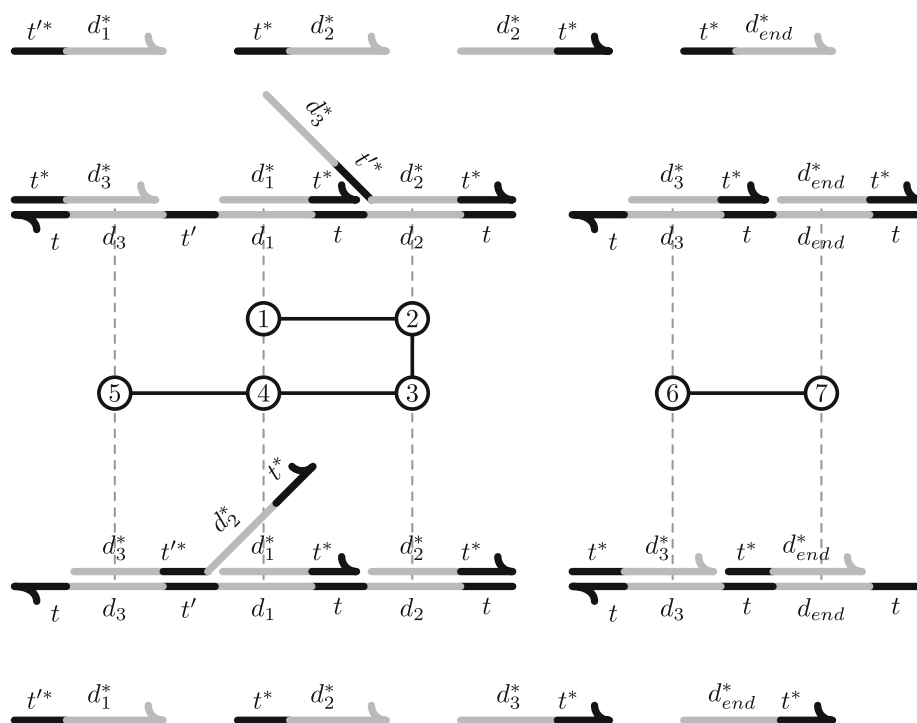
substring of the template, then the part of condition (5), which states that a signal binds to a template with exactly two adjacent domains, is implied. For example, consider a signal  $a^*b^*c^*d^*e^*$  and a template  $uvabdexy$ . Then a scattered substring  $a^*b^*d^*e^*$  of the signal of length four could bind to a substring  $abde$  of the template, thus breaking condition (5). The second part of condition (5), which states that a signal strand does not simultaneously bind to two different templates, is commonly assumed in any system which we have seen in the literature and is necessary for our proofs to work.

As a consequence of these conditions we have that the only way one configuration can be transformed to another configuration is through “positional strand displacement” described below, which for example, does not allow cooperative displacement, thus the name “uncooperative DSD” for our model. (As we will see later, a sequence of strand displacements can “walk” back and forth in templates, with each displacement using toeholds that become open as a result of the previous displacement in the sequence, but such walks are necessarily restricted to remain within a template. For example, see Fig. 6.)

–  $\mathcal{C}_{init}$  is an initial configuration.

Starting with the initial configuration, DSDs can progress through a sequence of configurations via positional strand displacements (PDs). PDs can move the open toehold of the template to the right or to the left. A PD moving the open toehold to the right is specified by a positive even number  $k$ , a template strand  $T$  with at least  $k + 1$  domains and a signal strand called the *invader*, say of type  $I$ , see Fig. 2a, where we can now assume that only positions  $k - 1$ ,  $k$ ,  $k + 1$  of template  $T$  are shown. The domain  $d$  at position  $k$  of the template is a long-domain and the domain at position  $k - 1$  is a toehold, say  $t$ . For the displacement to be applicable to a given configuration  $\mathcal{C}$ , it must be that in  $\mathcal{C}$  an additional signal strand, which we refer to as the *releasee*, is bound to  $d$  at position  $k$  and to a toehold at position  $k + 1$  of the template  $T$ , and the toehold at position  $k - 1$  is unbound (open). The invader is unbound in  $\mathcal{C}$  and contains the substring  $t^*d^*$ .

A displacement models the following steps in Fig. 2b,c,d, when toeholds and long-domains are actual DNA sequences. First, toehold  $t^*$  of the invader binds to the toehold  $t$  of the template at position  $k - 1$ . Then a branch migration ensues, whereby long-domain  $d^*$  of the invader binds to  $d$  at position  $k$  of the template and the releasee is no longer bound at this position. Finally, if it exists, the bond between the releasee and the toehold at position  $k + 1$  is broken. Thus in the resulting configuration  $\mathcal{C}'$ , substring  $t^*d^*$  of the invader is bound to  $td$  on the template at positions  $k - 1$  and  $k$  and the releasee is unbound, see Fig. 2e.



**Fig. 6** This figure shows the initial configuration  $C_{init}$  (top) and the final configuration (bottom) of a small UDSD. The UDSD has two templates and nine signals with one of them being a complex signal. The final configuration is characterized by the release of the signal  $s_{end} = d_{end}^*t^*$ . The middle portion of the figure illustrates the sequence of PDs that moves the UDSD from  $C_{init}$  to the release of  $s_{end}$ . The PDs are ordered according to the sequence in which they

occur, and the number for the PD is located horizontally over/under the template domain that is affected by the PD. The complex signal is active in two PDs, 2 and 5. PDs 1–5 interact with the first template by walking first to the right and then to the left along the template domains. This behavior is discussed in detail in Fig. 8. A more detailed illustration of the signals that participate in each PD is shown in Fig. 7

Formally a positional displacement (PD) of UDSD  $\Delta$  is a tuple of the form  $(I, T, k, z)$ , where  $I$  is a signal strand type,  $T$  is a template strand,  $k$  is a positive even integer and  $z \in \{L, R\}$ . PD  $(I, T, k, z)$  is *applicable* to a configuration  $C$  if the following conditions hold:

1. Strand  $T$  has at least  $k + 1$  domains and the  $k$ th domain, say  $d$ , is a long-domain. Also a signal strand, called the releasee, is bound to the  $k$ th domain of  $T$ .
2. In the configuration  $C$ , a strand of type  $I$  is unbound.
3. If  $z = R$  the following conditions hold. (Conditions for  $z = L$  are symmetric, with  $k + 1$  swapped with  $k - 1$  and  $d^*t^*$  replacing  $t^*d^*$ .)
  - (a) The  $(k - 1)$ st domain of  $T$  must exist and is a toehold, say  $t$ .
  - (b) A strand of type  $I$  contains substring  $t^*d^*$ .
  - (c) The releasee is also bound to a toehold at position  $k + 1$  of  $T$ . No other domains of the releasee are bound.
  - (d) The toehold at position  $k - 1$  of strand  $T$  is unbound. We call this toehold the *input toehold* of PD  $(I, T, k, z)$ .

The PD must *release* exactly one signal strand. Suppose that PD  $(I, T, k, z)$  is applicable to  $C$ . Let  $C'$  be obtained from  $C$  by removing the bonds between  $T$  and the releasee and by adding bonds either between any substring  $t^*d^*$  of an unbound strand of type  $I$  of  $C$  and the domains  $td$  at positions  $k - 1$  and  $k$  of  $T$  if  $z = R$ , or between any substring  $d^*t^*$  of  $I$  and the substring  $dt$  at positions  $k$  and  $k + 1$  of  $T$  if  $z = L$ . Then we say that  $(I, T, k, z)$  induces  $C'$  from  $C$ . This definition excludes *cooperativity* where two invading strands release a single releasee or one invading strand releases two releasees, because, by definition, every PD must be initiated by one invader and release exactly one releasee.

A sequence of PDs  $\rho = p_1, p_2, \dots, p_{|\rho|}$  is *valid* with respect to  $C_{init}$  if there is a sequence  $C_1, C_2, \dots, C_{|\rho|+1}$  of configurations of  $\Delta$  with  $C_1 = C_{init}$  such that for all  $i, 1 \leq i \leq |\rho|$ ,  $p_i$  is applicable to  $C_i$  and induces  $C_{i+1}$  from  $C_i$ . When  $C_{init}$  is clear from the context, we simply say that  $\rho$  is valid. A valid sequence *produces* a strand  $s \in \mathcal{S}$  if in  $C_{|\rho|+1}$ , the strand  $s$  is unbound. Let  $\text{Invaders}(\rho)$  be the set of types of invaders of  $\rho$ . Let  $\text{Unbound}(\rho, C_{init})$  be the set of types of unbound signals in  $C_{|\rho|+1}$  and  $\text{Unbound}(\rho)$  the set of types of unbound signals in  $C_1 \cup \dots \cup C_{|\rho|+1}$ . For

example, Fig. 6 shows an initial configuration (top) and a final configuration (bottom) for a PD. There are two templates and nine signals, one of which is a complex signal. Each configuration shows the signals bound to the templates and the unbound signals above them.

Let  $\rho = p_1, p_2, \dots, p_{|\rho|}$  be a sequence of PDs. The *template subsequence*  $\rho(T)$  is the subsequence of  $\rho$  with PDs of the form  $p_i = (I_i, T, k_i, z_i)$  where  $u < k_i < v$ .

The *volume* of UDSD  $\Delta$  is the number of domains in  $\mathcal{S}$ .

### 3.2 The upper bounds

First, we use the fact that a UDSD with simple signals can be simulated by a tagged CRN with volume that is polynomial in the volume of the UDSD, and thus we can use the bound in Theorem 1 to obtain the following result. If  $\Delta = (\mathcal{S}, C_{init})$  is a UDSD, we define  $\Delta^{(x)}$  to be the UDSD  $(x \cdot \mathcal{S}, x \cdot C_{init})$ , where  $x \cdot C_{init}$  denotes the configuration that contains  $x$  copies of each complex in  $C_{init}$ .

**Theorem 8** *Let  $\Delta$  be a UDSD with simple signals. Let  $B$  be the number of types of initially bound signal strands and  $D$  be the total number of long-domains of all templates. If  $\Delta$  can produce  $s_{end}$ , then  $\Delta^{((D+1)(2D+B+1))}$  can produce  $s_{end}$  via a sequence of at most  $2D(D+1)(2D+B)$  PDs.*

*Proof* By definition, if UDSD  $\Delta$  contains only simple signals, each template  $T$  has exactly  $s+1$  configurations, where  $v$  is the number of domains and  $s = (v-1)/2$  the number of long-domains of  $T$ , depending on the position of the open toehold. We denote these configurations by  $T_1, \dots, T_{s+1}$  for template  $T$ . Let  $T[i]$  be the domain at position  $i$  of  $T$ . Then each PD acting on the domain  $i$  of  $T$  can be expressed as follows as a chemical reaction:

$$T[2i-1]^*T[2i]^* + T_i \rightleftharpoons T[2i]^*T[2i+1]^* + T_{i+1}$$

where  $T[2i-1]^*T[2i]^*$  and  $T[2i]^*T[2i+1]^*$  are simple signal strands and where the notation  $*$  indicates that  $T[k]^*$  can bind to the template at position  $k$ . We express all PDs of  $\Delta$  as reversible chemical reactions above and construct the initial multiset of CRN  $\mathbf{C}$  as follows. Each initially unbound signal is added to the initial multiset of  $\mathbf{C}$ . For each template  $T$ , we add molecule  $T_i$  corresponding to the initial configuration of  $T$  to the initial multiset of  $\mathbf{C}$ . It is easy to see that the constructed CRN  $\mathbf{C}$  exactly simulates UDSD  $\Delta$ .

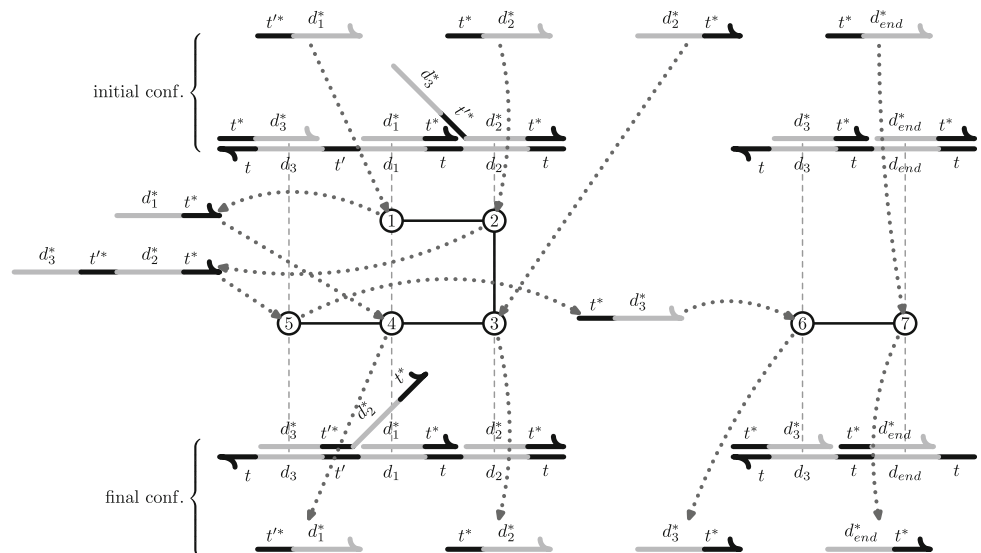
However, in order to apply the bound for CRNs from Sect. 2, we need to convert  $\mathbf{C}$  to a tagged CRN  $\mathbf{C}'$ . We express each PD acting on the domain  $T[2i]$  of  $T[u, v]$  as follows as a chemical reaction:

$$\tau_{T_i}^+ + T[2i-1]^*T[2i]^* + T_i \rightleftharpoons \tau_{T_i}^- + T[2i]^*T[2i+1]^* + T_{i+1}$$

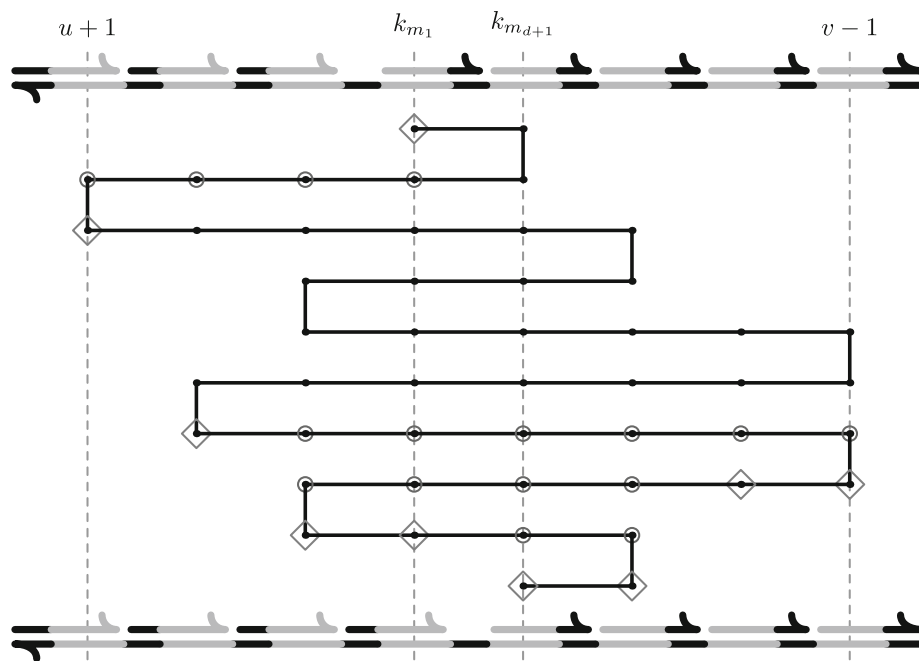
where  $\tau_{T_i}^+$  and  $\tau_{T_i}^-$  are unique tags of this reaction. The tagged CRN  $\mathbf{C}'$  exactly simulates CRN  $\mathbf{C}$  under the assumption that there are sufficiently many tags. Assume that the template  $T$  has  $t$  copies in  $\Delta$ . For one template, a single copy of each tag is enough to guarantee that the template can transform from one state to another. Therefore, we add  $t$  copies of tags  $\tau_{T_i}^+$  and  $\tau_{T_i}^-$  over all configurations  $T_i$  of all templates  $T$  to the initial multiset of tags of  $\mathbf{C}'$ . This number of tags is sufficient as it allows each simulated templates to freely transform between their configurations (assuming the required simulated signal strands are available). Note that the total number of tags added for one copy of domain  $T$  is exactly  $2s$ .

Finally, we need to determine the parameters of the constructed tagged CRN  $\mathbf{C}'$ . The number of types of signal molecules which are not initially present in the initial configurations, i.e.,  $|S| - |\llbracket S_0 \rrbracket|$ , is the number of types of

**Fig. 7** This figure shows more detail for the sequence of PDs outlined in Fig. 6. The initial configuration  $C_{init}$  is shown at the top and the final configuration with the release of signal  $s_{end} = d_{end}^*t^*$  is shown at the bottom. Each PD is indicated by a number in a circle where the numbers give the order in which the PDs occur. Each PDs circle has an incoming edge and an outgoing edge. The incoming edge indicates which signal is the invader and the outgoing edge indicates which signal is the release



**Fig. 8** An example of construction of the  $\beta_i(T)$  subsequence. At the top is the form of the initial configuration of the affected part of the template and at the bottom the final configuration. Each dot represents a PD of template subsequence  $\alpha_i(T)$ , each diamond a marked PD and each circle a connector PD. The sequence of PDs  $\beta_i(T)$  is then a subsequence of  $\alpha_i(T)$  which contains only the marked and connector PDs



initially bound signal strands in  $\Delta$  plus the sum of the numbers of configurations over all templates. Since the number of configurations of a template with  $s$  long-domains is  $s + 1$ , this number can be upper bounded by  $B + 2D$ . The number of tags is exactly  $2D$ . The bandwidth of  $C'$  is 1. The theorem follows by Theorem 1.  $\square$

As shown in Fig. 3, the proof of Theorem 8 will not work in the case of general signal strands, since the number of configurations of some templates can be exponential. Instead of simulating a UDSD by a tagged CRN, in Theorem 9 we will prove a bound for general (i.e., not with simple signals) UDSDs directly, reusing some ideas of the proof for tagged CRNs.

Let  $\Delta$  be a UDSD. Our goal is to show that if there is a valid sequence of PDs  $\alpha = q_1, q_2, \dots, q_{|\alpha|}$  that produces a given signal  $s_{end}$  in  $\Delta$ , for example Fig. 6, then there is a “shorter” valid sequence,  $\gamma$ , that produces  $s_{end}$  in a multi-copy version of  $\Delta$ , i.e., a version that initially has many copies of  $C_{init}$ . Moreover, the number of copies of  $C_{init}$  and the length of  $\gamma$  will be bounded by a polynomial in  $B$ , the number of types of signals that are initially-bound (i.e., every copy is bound) in  $C_{init}$  but are released by  $\alpha$ ; and  $D$ , the total number of long-domains of all templates. We first provide some intuition for our proof while introducing some useful definitions, and then provide the formal details in a series of claims.

To build intuition for our proof, we present three possible strategies for constructing  $\gamma$ . The first two strategies are flawed but provide motivation for the details of the third, correct, strategy.

**Strategy 1:** Let  $\gamma$  be the sequence of PDs that, starting from the initially open toehold of a template in which  $s_{end}$  is bound, “walks”, i.e., displaces the bound signals one at a time, between this open toehold and  $s_{end}$ . For example, in Fig. 6, the signal  $t^*d_3^*$  would be used to initiate a sequence of PDs starting at the left of the second template and finally releasing  $s_{end}$  at the far right.

The  $\gamma$  of Strategy 1 has length at most  $D$ . However, the multiset of invader signals needed for the displacements may not be in  $C_{init}$ . To overcome this problem, we need  $\gamma$  to release (enough copies of) each signal that is not in  $C_{init}$  but that is released by  $\alpha$ . For each type  $s$  of signal strand in Unbound ( $\alpha$ ) that is bound in  $C_{init}$ , let  $\text{index}(s)$  be the index of the first PD of  $\alpha$  that releases  $s$ . Let  $s_1, \dots, s_B (= s_{end})$  be the sequence of all such signals ordered by their indexes, i.e.,  $\text{index}(s_1) < \text{index}(s_2) < \dots < \text{index}(s_B)$ , until  $s_{end}$  is produced. Let  $S_i = \{s_1, \dots, s_i\}$ . Let  $\alpha_i = q_1, q_2, \dots, q_{\text{index}(s_i)}$ . For example in Fig. 6,  $s_1$  is the signal  $d_1^*t^*$  and  $s_{end}$  is  $d_{end}^*t^*$ .

**Strategy 2:** Using Strategy 1, and taking advantage of the fact that multiple copies of  $C_{init}$  are available initially, the PDs in  $\gamma$  first produce (sufficiently many) copies of signal  $s_1$ . This is possible because by definition of  $s_1$ , there is a walk of length at most  $D$  to some  $s_1$  that only uses invaders in  $C_{init}$ . In a similar manner, use signals in yet additional copies of  $C_{init}$  plus the newly released signals  $s_1$  to release signal  $s_2$ , and so on. For example in Fig. 6,  $s_1$  is the signal  $d_1^*t^*$ ,

and by using multiple copies of the initial configuration, we can get copies of  $s_1$  which help us get copies of  $s_2$ , etc.

The problem with this strategy is that the number of copies of  $\mathcal{C}_{init}$  available initially may need to be exponential in  $D$ , in order to release  $s_B (= s_{end})$ . Specifically,  $\Theta(D)$  copies of  $\mathcal{C}_{init}$  would be needed to produce one copy of  $s_1$ , e.g., in a scenario where all of the needed invaders on the walk to  $s_1$  are identical, there is only one copy of this invader in  $\mathcal{C}_{init}$  and  $s_1$  has distance  $\Theta(D)$  from the initially free toehold in its template. Thus we would need  $\Theta(DX)$  copies of  $\mathcal{C}_{init}$  to produce  $X$  copies of  $s_1$  using Strategy 2. By the same argument we may need  $\Theta(DX)$  copies of  $(\mathcal{C}_{init} \cup s_1)$  to get  $X$  copies of  $s_2$ , leading to a total of  $\Theta(D^2X)$  copies of  $\mathcal{C}_{init}$  to produce  $X$  copies of  $s_2$ , and so on. To overcome this problem, we need  $\gamma$  to take a walk that, while still being short, is more effective in releasing needed invaders and more conservative about using them up.

**Strategy 3:** This strategy first releases a copy of  $s_1$  via a short walk  $\beta_1$  that uses invaders from just a single copy of  $\mathcal{C}_{init}$  but that can also “borrow” signals from a reserve of extra copies of  $\mathcal{C}_{init}$ , as long as the signals are returned to the reserve by the end of the walk. For example in Fig. 6, we would have many copies of the initial configuration (top) which give us a reserve of signals.

We construct  $\beta_1$  by adapting  $\alpha_1$  (the prefix of  $\alpha$  that causes  $s_1$  to be released, see above). Note that  $\alpha_1$  releases  $s_1$  without needing to borrow from a reserve, but may be too long for our result. In contrast,  $\beta_1$  will have length  $O(D^2)$  and at the same time, the set of initially-bound signals that are released by  $\beta_1$  will be the same as that of  $\alpha_1$  and the set of signals that are finally-bound by  $\beta_1$  will also be the same as that of  $\alpha_1$ . In particular, all other signals, e.g., from the reserve, that may temporarily be bound during the walk taken by  $\beta_1$  are also released during the walk. The sequence of PDs  $\beta_1$  sweeps across the region traversed by  $\alpha_1$  in a zig-zag fashion (for a specific example, see Fig. 6), so as to visit each domain for (Fig. 7) the *last* time in the same order as does  $\alpha_1$ . Figure 8 provides a general example. When visiting a domain for the last time,  $\beta_1$  uses the last PD of  $\alpha_1$  that visits that domain—these PDs are called *marked* PDs in the formal description to come later. This ensures that the set of signals that are finally-bound by  $\beta_1$  is the same as that of  $\alpha_1$ . Also in  $\beta_1$ , between the marked PDs, are intermediate “connector” PDs that ensure that  $\beta_1$  is a valid sequence of PDs. The connector PDs are also chosen so that the *first* PD of  $\beta_1$  that releases a signal at a given domain is the same as the first PD

of  $\alpha_1$  that releases a signal at that domain. This ensures that the set of initially-bound signals that are released by the end of  $\beta_1$  is the same as that of  $\alpha_1$ .

The walk  $\beta_1^X$  can produce  $X$  copies of  $s_1$  using  $X$  copies of  $\mathcal{C}_{init}$  plus a “reserve” of  $|\beta_1|$  copies of  $\mathcal{C}_{init}$  that is still available at the end of the walk. In a similar fashion, copies of  $s_2$  can then be produced by consuming one additional copy of  $\mathcal{C}_{init}$  per copy of  $s_2$ , and also borrowing from the growing reserve of signals, namely multiple copies of all signals in  $\mathcal{C}_{init} \cup \{s_1\}$ . Continuing in this way,  $s_B = s_{end}$  can be generated from an initial number of copies of  $\mathcal{C}_{init}$  that is bounded by a polynomial in  $B$  and  $D$ .

We now present the formal details. Let  $T$  be a template, and let  $\alpha_i(T) = p_1, p_2, \dots, p_{|\alpha_i(T)|}$  be the template subsequence of  $\alpha_i$ , where  $p_j = (I_j, T, k_j, z_j)$  for every  $j = 1, \dots, |\alpha_i(T)|$ . Let  $u$  and  $v$  the first and last toeholds of  $T$  affected by  $\alpha_i(T)$ , respectively, and  $d = (v - u)/2$  the number of affected long-domains in  $T$ . We construct a subsequence  $\beta_i(T)$  of the PDs in  $\alpha_i(T)$ . The PDs in this subsequence will be of two types, *marked* and *connector*.

### 3.2.1 Marked PDs

Mark the first PD  $p_1$  of  $\alpha_i(T)$ , and then mark, for *each* affected long-domain in the template  $T$ , the *last* PD of  $\alpha_i(T)$  that binds to it. Let  $p_{m_1}, \dots, p_{m_{d+1}}$  be the subsequence of all marked PDs ( $1 = m_1 < m_2 < \dots < m_{d+1}$ ). It is easy to see that the sequence of marked PD positions,  $k_{m_2}, \dots, k_{m_d}$ , consists of two interleaved monotonic subsequences:  $U = u + 1, u + 3, \dots, k_{m_{d+1}} - 2$  and  $V = v - 1, v - 3, \dots, k_{m_{d+1}} + 2$ , where  $k_{m_{d+1}}$  is the long-domain position of the last PD in  $\alpha_i(T)$ . Furthermore, the marked PDs with the long-domains in the first subsequence have direction  $R$  and in the second subsequence direction  $L$ . Depending on the direction  $z_{m_{d+1}}$  of the last marked PD, we add the long-domain position  $k_{m_{d+1}}$  at the end of  $U$  if  $z_{m_{d+1}} = R$  or at the end of  $V$ , if  $z_{m_{d+1}} = L$ .

### 3.2.2 Connector sequences

Now, we must connect the marked PDs by introducing *connector sequences* of PDs between each consecutive pair of marked PDs with the goal being for each subsequent PD to use the toehold opened by the previous PD. Let  $\bar{z}$  indicate the opposite direction from  $z$ .

For the connector sequence connecting  $p_{m_1}$  and  $p_{m_2}$ , select as a connector the *first* PD in  $\alpha_i(T)$  with direction  $\bar{z}_{m_2}$  that binds to each long-domain of  $T$  between positions  $k_{m_2}$  and  $k_{m_2}$  inclusive. It is easy to see that either all selected connector PDs are before  $p_{m_2}$  in the sequence  $\alpha_i(T)$ , or

$m_1 = m_2$  and the connector sequence is empty. In the second case,  $k_{m_1}$  is either  $u + 1$  or  $v - 1$ , and there is no other PD in  $\alpha_i(T)$  with the same long-domain position.

Consider  $j = 2, \dots, d$ . Each PD of the connector sequence connecting  $p_{m_j}$  to  $p_{m_{j+1}}$  will be between  $p_{m_j}$  and  $p_{m_{j+1}}$  in the sequence  $\alpha_i(T)$ . We will consider two cases.

1. If  $z_{m_j} = z_{m_{j+1}}$ , then no connector PDs are needed (long-domain positions  $k_{m_j}$  and  $k_{m_{j+1}}$  are from the same subsequence—either  $U$  or  $V$ —and hence they differ by exactly 2).
2. If  $z_{m_j} \neq z_{m_{j+1}}$ , then we select the connectors as follows. In the subsequence of  $\alpha_i(T)$  between PDs  $p_{m_j}$  and  $p_{m_{j+1}}$ , choose as a connector the *first* PD that binds to each position between  $k_{m_j}$  and  $k_{m_{j+1}}$ , excluding position  $k_{m_j}$  and including position  $k_{m_{j+1}}$ . Note that each PD in this connector sequence must have direction  $z_{m_j}$ .

The construction is illustrated in Fig. 8. The sequence  $\beta_i(T)$  contains all the marked PDs and all the connector PDs, with distinct indices. Note that this is a subsequence of  $\alpha_i(T)$  since for every  $j = 1, \dots, d$ , the connector sequence connecting  $p_{m_j}$  to  $p_{m_{j+1}}$  contains only PDs between  $p_{m_j}$  and  $p_{m_{j+1}}$ . Finally, we define  $\beta_i$  as a concatenation of  $\beta_i(T)$ 's over all templates  $T$  in  $C_{init}$ .

We next state and prove a sequence of claims that we use to prove our main result.

**Claim 2** *The first PD in the sequence  $\beta_i(T)$  can use the initially open toehold. Every other PD in the sequence can use the toehold opened by the previous PD in the sequence.*

*Proof* The first part of the claim is straightforward since the first PD of  $\beta_i(T)$ , i.e.,  $p_{m_1}$ , is also the first PD of  $\alpha_i(T)$ .

For the second part of the claim, first, we note that each connector sequence connecting two consecutive marked PDs consists of PDs with the same direction such that for all two consecutive PDs in the connector sequence, their long-domain positions differ by the same value in  $\{-2, 2\}$ . Therefore, all but the first PD in the connector sequence uses a toehold open by the previous PD in the sequence. It is enough to show that this condition is satisfied also

- (a) between the first PD of a connector sequence and the preceding marked PD; and
- (b) between the last PD of a connector sequence and the following marked PD,

for each non-empty connector sequence. In the case that some connector sequence is empty, we need to check that the condition is satisfied between the marked PDs preceding and following the empty connector sequence. Consider the later case first: assume that the connector sequence connecting  $p_{m_j}$  to  $p_{m_{j+1}}$  is empty. Then  $k_{m_j}$  and  $k_{m_{j+1}}$  belong to the same

monotonic subsequence (either  $U$  or  $V$ ), i.e.,  $k_{m_{j+1}} = k_{m_j} \pm 2$ , and hence,  $p_{m_{j+1}}$  uses the toehold opened by  $p_{m_j}$ .

Consider the connector sequence connecting  $p_{m_1}$  to  $p_{m_2}$ . If  $m_1 = m_2$ , both conditions are trivially satisfied. Otherwise, PD  $p_{m_1}$  may or may not be in this connector sequence. If  $p_{m_1}$  is a connector, the first PD of the connector sequence is  $p_{m_1}$ , hence condition (a) is trivially satisfied. If PD  $p_{m_1}$  is not a connector then  $z_{m_1} = z_{m_2}$  and either the connector sequence is empty, or the long-domain position of the first PD of the connector sequence is  $k_{m_1}$ , that is, the first PD of the connector sequence uses the toehold opened by  $p_{m_1}$ , condition (a) holds. The long-domain position of the last PD of the non-empty connector sequence is  $k_{m_2}$  and the direction of the last PD is  $\bar{z}_{m_2}$ , hence condition (b) is satisfied.

Next, consider the connector sequence connecting  $p_{m_j}$  to  $p_{m_{j+1}}$ , for  $j = 2, \dots, d$ .

1. If  $z_{m_j} = z_{m_{j+1}}$ , then the connector sequence is empty.
2. If  $z_{m_j} \neq z_{m_{j+1}}$ , then the long-domain position of the first PD in the connector sequence is  $k_{m_j} + 2$  if  $z_{m_j} = R$  and  $k_{m_j} - 2$  if  $z_{m_j} = L$ . In either case, the condition (a) is satisfied. Furthermore, the long-domain position of the last PD of the connector sequence is  $k_{m_{j+1}}$  and its direction is  $\bar{z}_{m_{j+1}}$ , hence, condition (b) is satisfied.  $\square$

**Claim 3** *The length of  $\beta_i(T)$  is at most  $(\delta + 1)(\delta + 2)/2$ , where  $\delta$  is the number of long-domains in  $T$ .*

*Proof* Let  $u$  and  $v$  be the first and last affected toeholds of  $T$  by  $\beta_i(T)$  and  $d = (v - u)/2 \leq \delta$  the number of affected long-domains. The number of marked PDs is  $d + 1$ . The first connector sequence has at most  $d$  PDs. For each  $j = 2, \dots, d$ , consider the connector sequence connecting marked PDs  $p_{m_j}$  and  $p_{m_{j+1}}$ . If  $z_{m_j} = z_{m_{j+1}}$ , the sequence is empty. Otherwise, the number of PDs in the sequence is at most  $|k_{m_{j+1}} - k_{m_j}|/2$  and  $k_{m_j}$  and  $k_{m_{j+1}}$  belong to different monotonic subsequences of positions. Without loss of generality, assume  $k_{m_j}$  is at index  $r$  in  $U$  and  $k_{m_{j+1}}$  is at index  $r'$  in  $V$ . Since each marked PD advances by one element in exactly one of the sequences  $U$  and  $V$ , we have  $r + r' = j$ , and therefore  $|k_{m_{j+1}} - k_{m_j}|/2 = |[v - (2r' - 1)] - [u + (2r - 1)]|/2 = |v - u - 2j + 1|/2 = d - j + 1$  (the last equality follows since  $j \leq d$ ). Hence, the number of connector PDs is at most  $d + \sum_{j=2}^d (d - j + 1) = d(d + 1)/2$  and the total number of PDs in  $\beta_i(T)$  as at most  $(d + 1)(d + 2)/2 \leq (\delta + 1)(\delta + 2)/2$ .  $\square$

**Claim 4** *The length of  $\beta_i$  is at most  $(D + 1)(D + 2)/2$  and thus  $|\text{Invaders}(\beta_i)| \leq (D + 1)(D + 2)/2$ . Also,  $\text{Invaders}(\beta_i)$  contains only types of unbound strands of  $C_{init}$  or strand types in  $S_{i-1} = \{s_1, \dots, s_{i-1}\}$ .*

*Proof* By Claim 3, for each template  $T$ , the number of PDs of  $\beta_i(T)$  is at most  $(\delta + 1)(\delta + 2)/2$ , where  $\delta$  is the number of long-domains in  $T$ . Summing through all domains, we obtain that the length of  $\beta_i$  is at most  $(D + 1)(D + 2)/2$ .

By definition of  $\alpha_i$ ,  $\text{Invaders}(\alpha_i)$  contains only types of unbound strands of  $C_{init}$  or strand types in  $\{s_1, \dots, s_{i-1}\}$ . Since  $\beta_i$  is a subsequence of  $\alpha_i$ , it must also be that  $\text{Invaders}(\beta_i)$  also contains only types of unbound strands of  $C_{init}$  or strand types in  $S_{i-1}$ .  $\square$

**Claim 5**  $\beta_i$  is valid with respect to

$$C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1}).$$

Moreover,

$$(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1}) \subseteq \text{Unbound}(\beta_i, C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})).$$

*Proof* Let  $\beta_i = p'_1, p'_2, \dots, p'_{|\beta_i|}$ . To prove the first part of the claim, we need to show that there is a sequence  $C_1, C_2, \dots, C_{|\beta_i|+1}$  of configurations with  $C_1 = C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})$  such that for all  $j, 1 \leq j \leq |\beta_i|, p'_j$  is applicable to  $C_j$  and induces  $C_{j+1}$  from  $C_j$ . We can prove this by induction on  $j$ . The base case when  $j = 1$  is trivial. Suppose that  $j > 1$ , and that  $p'_{j-1}$  is applicable to  $C_{j-1}$  and induces  $C_j$  from  $C_{j-1}$ . Let  $p'_j = (I, T, k, z)$ . Since  $(I, T, k, z)$  is also a PD of  $\alpha_i$  and  $\alpha_i$  is valid, it is straightforward to check that condition 1 of the definition of “applicable” must hold. Condition 2 also holds because  $j \leq (D + 1)(D + 2)/2$  and there are  $(D + 1)(D + 2)/2$  copies of all unbound signals used by  $\beta_i$  initially present in  $(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})$ . So, we assume that  $z = R$  and show that condition 3 holds (the argument is similar when  $z = L$ ). Condition 3a and 3b also follow simply from the fact that  $(I, T, k, z)$  is a PD of  $\alpha$ .

Condition 3c, that the releasee is not bound to any domain except the neighboring toehold, must be true because  $(I, T, k, z)$  is a PD of  $\alpha$ . The condition 3d follows by Claim 2.

The final multiset of signals after executing PDs in  $\alpha_i$  on  $C_{init}$  is the multiset consisting of all signals of  $C_{init}$  plus all signals initially bound to domains that appear in PDs of  $\alpha_i$  minus all signals that are finally bound to domains that appear in PDs of  $\alpha_i$ . By construction, PDs in  $\beta_i$  operate on exactly the same set of long-domains as PDs in  $\alpha_i$  and the last PD applied to each long-domain of  $\alpha_i$  is exactly the same as those of  $\beta_i$ . Therefore, no matter whether we execute PDs in  $\alpha_i$  or PDs in  $\beta_i$ , exactly the same set of signals are released and bound, and hence, the final multiset of unbound signals is the same as well. It follows that

$$(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1}) \subseteq \text{Unbound}(\beta_i, C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})). \quad \square$$

**Claim 6** Let  $\beta_i^{(D+1)(D+2)/2}$  denote the sequence  $\beta_i$  concatenated  $(D + 1)(D + 2)/2$  times, modified just so that the PDs of each copy refer to templates of different copies of  $(D + 1)(D + 2)/2 \cdot C_{init}$ . Then  $\beta_i^{(D+1)(D+2)/2}$  is valid with respect to the configuration

$$(D + 1)(D + 2)/2 \cdot C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1}).$$

Moreover,

$$(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_i) \subseteq \text{Unbound}(\beta_i^{(D+1)(D+2)/2}, (D + 1)(D + 2)/2 \cdot C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})) \quad (1)$$

*Proof* By Claim 5,  $\beta_i$  is valid with respect to  $C_{init} \cup (D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})$ . Moreover, the final multiset of signals is the same as if we were to execute PDs in  $\alpha_i$  on  $C_{init}$  and then add  $(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})$ . Thus, if we repeat  $\beta_i$   $(D + 1)(D + 2)/2$  times and execute each  $\beta_i$  on a different copy of  $C_{init}$ , we will still have at least  $(D + 1)(D + 2)/2$  copies of signals in  $C \cup S_{i-1}$  plus  $(D + 1)(D + 2)/2$  copies of  $s_i$ .  $\square$

The proof of our main technical result, Theorem 9, follows from the preceding claim.

**Theorem 9** Let  $\Delta$  be a UDSD with  $B$  types of initially bound signal strands and let  $D$  be the total number of long-domains of all templates. If  $\Delta$  can produce  $s_{end}$ , then  $\Delta^{((D+1)(D+2)(B+1)/2)}$  can produce  $s_{end}$  via a sequence of at most  $(D + 1)^2(D + 2)^2B/4$  PDs.

*Proof* Let  $\alpha, \alpha_i$  and  $\beta_i, 1 \leq i \leq B$  be defined as above. Let  $\gamma$  be the sequence of PDs obtained by concatenating  $(D + 1)(D + 2)/2$  copies of sequence  $\beta_1$  followed by  $(D + 1)(D + 2)/2$  copies of  $\beta_2$  and so on up to  $(D + 1)(D + 2)/2$  copies of  $\beta_B$ , and modifying each copy just so that the PDs of each copy refer to templates of different copies of  $(D + 1)(D + 2)B/2 \cdot C_{init}$ .

By applying Claim 6, we will show by induction on  $i$  that

$$\beta_1^{(D+1)(D+2)/2} \beta_2^{(D+1)(D+2)/2} \dots \beta_i^{(D+1)(D+2)/2}$$

is valid with respect to  $(D + 1)(D + 2)(i + 1)/2 \cdot C_{init}$  and that

$$(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_i) \subseteq \text{Unbound}(\beta_1^{(D+1)(D+2)/2} \beta_2^{(D+1)(D+2)/2} \dots \beta_i^{(D+1)(D+2)/2}, (D + 1)(D + 2)(i + 1)/2 \cdot C_{init}). \quad (2)$$



It follows that  $\gamma$  is valid with respect to  $(D + 1)(D + 2)(B + 1)/2 \cdot C_{init}$  and that  $\gamma$  produces  $s_{end}$  ( $=s_B$ ). The base case is when  $i = 1$ . It follows directly from Claim 6 that  $\beta_1^{(D+1)(D+2)/2}$  is valid with respect to the configuration  $(D + 1)(D + 2) \cdot C_{init}$  and that  $(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_1) \subseteq \text{Unbound}(\beta_1^{(D+1)(D+2)/2}, (D + 1)(D + 2) \cdot C_{init})$ . The induction hypothesis is that  $\beta_1^{(D+1)(D+2)/2} \beta_2^{(D+1)(D+2)/2} \dots \beta_{i-1}^{(D+1)(D+2)/2}$  is valid with respect to  $(D + 1)(D + 2)(i + 1)/2 \cdot C_{init}$  and that

$$(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1}) \subseteq \text{Unbound}(\beta_1^{(D+1)(D+2)/2} \beta_2^{(D+1)(D+2)/2} \dots \beta_{i-1}^{(D+1)(D+2)/2}, (D + 1)(D + 2)(i + 1)/2 \cdot C_{init}) \quad (3)$$

The induction step then follows easily from Claim 6 because after the PDs in

$$\beta_1^{(D+1)(D+2)/2} \beta_2^{(D+1)(D+2)/2} \dots \beta_{i-1}^{(D+1)(D+2)/2}$$

have been applied on  $(D + 1)(D + 2)(i + 1)/2 \cdot C_{init}$ , the resulting configuration includes both  $(D + 1)(D + 2)/2 \cdot (C_{init} \cup S_{i-1})$  (by the induction hypothesis) and  $(D + 1)(D + 2)/2$  additional copies of  $C_{init}$  needed for sequence  $\beta_i^{(D+1)(D+2)/2}$ . Also, each of the  $(D + 1)(D + 2)/2$   $\beta_i$  sequences produces a single  $s_i$  that remains unbound when the remaining  $\beta_i$ 's are applied.  $\square$

Finally, we restate Theorem 9 for copy-tolerant UDSDs. We say that a UDSD is  $x$ -copy-tolerant if the length of the shortest PD sequence that produces any signal strand  $s$  in  $\Delta$  and in  $\Delta^{(x)}$  is the same. A UDSD is copy-tolerant if it is  $x$ -copy-tolerant for all  $x$ .

**Theorem 10** *Let  $\Delta$  be a UDSD with  $B$  types of initially bound signal strands and let  $D$  be the total number of long-domains of all templates. If  $\Delta$  can produce  $s_{end}$  and  $\Delta$  is  $(D + 1)(D + 2)(B + 1)/2$ -copy tolerant, then  $\Delta$  can produce  $s_{end}$  via a sequence of at most  $(D + 1)^2(D + 2)^2B/4$  PDs.*

### 3.2.3 Concatenated templates

The result above may seem to be limited, due to the definition of templates beginning and ending with toehold domains and consisting of alternating toehold and long-domains. Let a *generalized template* consist of several templates concatenated together. In fact, the result as stated in Theorem 9 also applies to generalized templates. This is because a UDSD with generalized templates can be simulated w.l.o.g. by a UDSD with a sufficient number of templates. This makes our result more general.

### 3.2.4 Irreversible reactions

If irreversible reactions are considered, then we must allow for there to be no toehold to either the left or right of a long-domain on the template. If we are to keep the condition that every PD has a releasee, then we must allow for some releasee to contain only the long-domain, rather than the toehold. This complicates the proofs, because the current development maintains that at any time there is only one open toehold in each template. In order to both allow irreversible reactions and use the current proofs, we must require that in  $C_{init}$  one-domain releasees only appear where there is no toehold to either the right or the left. Because we find this restriction somewhat artificial, we conjecture that there is some generalization of these proof that allows for irreversible reactions.

## 4 Conclusions and open questions

In this paper, we have considered three models of biomolecular programs, namely tagged CRNs, DSDs, and DSDs with simple signals. We have shown that, when multiple copies of all initial molecules are present, such programs fail to work correctly if the number of reactions of the program is sufficiently large relative to the volume of initial reagents. A natural question is: how do these models relate to each other, in the sense that one can be simulated by another? Soloveichik et al. showed how CRNs (and thus also tagged CRNs) could be simulated by DSDs, in the sense that CRN species are mapped to DSD signals, CRN reactions can be simulated by a cascade of DSD strand displacements, and the dynamical properties of the CRN are reproduced. As a consequence, programs specified as CRNs can be compiled into real, DNA-based chemical systems and there are several examples to date. We are not aware of general methods for simulating DSDs by CRNs. Such simulations are possible in principle, for example by mapping each multi-stranded complex that could arise in a configuration of the DSD to be simulated, to distinct abstract species of the simulating CRN. However, the number of species could be exponential in the size of the DSD, and it's not clear what purpose such a simulation would serve.

There are many open questions about the potential for CRNs and DSDs to be correct in the multi-copy setting. First, can our reachability upper bound results be strengthened? There are two possible ways to strengthen our result for CRNs (Theorem 2): either by reducing the length of the shortest computation needed to produce  $s_{end}$

or to show that the system is not  $x$ -copy tolerant for some  $x < |S|b_C(T_C/2 + 1)$ . Similarly, there are two ways to strengthen the reachability upper bounds for DSDs.

Also, can our result on DSDs be extended to DSDs with more complex primitives, such as cooperative strand displacement (Zhang 2011) or irreversible reactions? What if long-domains can form intra-molecular bonds, e.g., forming hairpins, in addition to inter-molecular bonds?

This paper considers only reachability bounds, i.e., bounds on the number of reactions (steps) needed to reach (produce) a given product. However, real CRNs behave stochastically, with rates that depend on relative quantities of species. It is plausible that the lack of robustness implied by our theorems, i.e., errors that occur in the multi-copy setting in CRNs that fail to satisfy the conditions of the theorem, would be very unlikely to occur in some CRNs and thus would not be an issue in a real system. Analyses of robustness of CRNs under stochastic assumptions, perhaps computing expected hitting times, would help us better understand the degree to which robustness issues are a problem.

**Acknowledgments** We would like to thank the anonymous referee who helped us to improve the bound in Theorem 7 and inspired us to construct a CRN which shows that this bound is tight (Example 2).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

## References

- Cardelli L (2010) Two-domain DNA strand displacement. In: Proceedings of developments in computational models (DCM 2010). Electronic proceedings in theoretical computer science, vol 26, pp 47–61
- Chen HL, Doty D, Soloveichik D (2012) Deterministic function computation with chemical reaction networks. In: DNA computing and molecular programming—18th international conference. Lecture Notes in Computer Science, vol 7433. Springer, Berlin, pp 25–42
- Condon A, Hu AJ, Mañuch J, Thachuk C (2012) Less haste, less waste: on recycling and its limits in strand displacement systems. *J R Soc Interface*
- Cook M, Soloveichik D, Winfree E, Bruck J (2009) Programmability of chemical reaction networks. *Algorithmic Bioprocess* 133:543–584
- Qian L, Soloveichik D, Winfree E (2011) Efficient turing-universal computation with DNA polymers. In: Proceedings of the sixteenth annual conference on DNA computing and molecular programming. Lecture Notes in Computer Science, vol 6518. Springer, Berlin, pp 123–140
- Qian L, Winfree E (2011) Scaling up digital circuit computation with DNA strand displacement cascades. *Science* 332:1196–1201
- Qian L, Winfree E, Bruck J (2011) Neural network computation with DNA strand displacement cascades. *Nature* 475:368–372
- Seelig G, Soloveichik D, Zhang DY, Winfree E (2006) Enzyme-free nucleic acid logic circuits. *Science* 314(5805):1585–1588
- Soloveichik D (2009) Robust stochastic chemical reaction networks and bounded tau-leaping. *J Comput Biol* 16(3):501–522
- Soloveichik D, Cook M, Winfree E, Bruck J (2008) Computation with finite stochastic chemical reaction networks. *Nat Comp* 7:615–633
- Soloveichik D, Seelig G, Winfree E (2010) DNA as a universal substrate for chemical kinetics. *Proc Natl Acad Sci USA* 107(12):5393–5398
- Thachuk C, Condon A (2012) Space and energy efficient computation with DNA strand displacement systems. In: DNA computing and molecular programming—18th international conference. Lecture Notes in Computer Science vol 7433. Springer, Berlin, pp 135–149
- Yurke B, Mills AP (2003) Using DNA to power nanostructures. *Genet Program Evolvable Mach* 4(2):111–122
- Yurke B, Turberfield AJ, Mills AP, Simmel FC, Neumann JL (2000) A DNA-fuelled molecular machine made of DNA. *Nature* 406:605–608
- Zhang DY (2011) Cooperative hybridization of oligonucleotides. *J Am Chem Soc* 133:1077–1086
- Zhang DY, Seelig G (2011) Dynamic DNA nanotechnology using strand displacement reactions. *Nat Chem* 3:103–113
- Zhang DY, Turberfield AJ, Yurke B, Winfree E (2007) Engineering entropy-driven reactions and networks catalyzed by DNA. *Science* 318:1121–1125