Equivalence in Template-Guided Recombination*

Michael Domaratzki

Department of Computer Science
University of Manitoba
Winnipeg, MB R3T 2N2 Canada
mdomarat@cs.umanitoba.ca

Abstract. We consider theoretical properties of the template-guided recombination operation. In particular, we consider the decidability of whether two sets of templates are equivalent, that is, whether their action is the same for all operands. We give a language-theoretic characterization of equivalence which leads to decidability results for common language classes. In particular, we show a positive answer for regular sets of templates. For context-free sets of templates, the answer is negative.

1 Introduction

The rearrangement of DNA in stichotrichous ciliates has received a significant amount of attention in the literature as a model of natural computing. Several potential formal models for the rearrangement have been proposed, including both intra-molecular and inter-molecular models. Ehrenfeucht *et al.* [7] give a detailed overview of ciliate DNA rearrangement and an investigation of one of the proposed models.

Template-guided recombination (TGR) is one of the formal models for recombination of DNA in stichotrichs. The model, proposed by Prescott *et al.* [9], has been the subject of much research in the literature [3–6, 8]. Much of this work on TGR has focused on examining the closure properties of the operation. For example, McQuillan *et al.* [8] have recently shown that if a context-free language is iteratively operated upon with a regular set of templates (see Section 2 for definitions), then the resulting language is a context-free language which can be effectively constructed.

TGR specifies a set of templates which defines how the operation works: changes to the set of templates affect how the TGR operation functions on its operand, which represents the scrambled DNA in the ciliate. It is reasonable, therefore, to ask exactly what changes to the set of templates affect the operation of TGR. This is the question we address in this paper: given two sets of templates, do they define equivalent TGR operations? We give a natural condition on subwords of templates which exactly characterizes equivalence for template sets over an alphabet of at least three symbols.

From this characterization, we then establish decidability results: given two regular sets of templates, it is decidable whether they are equivalent. We also

* Research supported in part by a grant from NSERC.

show an interesting universality result: determining whether a set of templates is equivalent to the universal set of all templates is not difficult, as it can be decided even for recursive sets. However, for alphabets of size at least three, there exists a fixed regular set of templates T_0 such that it is undecidable if a given context-free set of templates is equivalent to T_0.

As a proposed model for natural computing, understanding the equivalence of template sets is a critical prerequisite for understanding the potential for employing the natural computing power of ciliate DNA rearrangement. Under the hypothesis that a distinct set of DNA material (the templates) exactly guides rearrangement, a potential method for altering of the computational action of the rearrangement is a modification of the set of templates which are present during rearrangement.

With the results in this paper, we are able to determine exactly the situations in which modifying the set of templates modifies the computational process of rearrangement which occurs. For a recent survey of experimental results and hypotheses in identifying exogenic factors affecting ciliate DNA rearrangement, see Cavalcanti and Landweber [2]. Recent experimental results lend some support to the TGR model. Vijayan *et al.* [11] demonstrate that the addition of permuted RNA to the parental macronucleus does affect the rearrangement process during conjugation, and a modified micronucleus is produced.

Recently, Angeleska *et al.* [1] have reconsidered the TGR model by incorporating RNA templates (either single-stranded or double-stranded RNA). Their model does not incorporate any part of the RNA template into the rearranged DNA and reduces the number of required cuts to the DNA backbones. However, as noted by the authors, the new model does not have any impact when considered as an inter-molecular operation of formal languages as we do here.

2 Preliminary Definitions

We use the tools of formal language theory to study TGR. For additional background on formal languages, see Rozenberg and Salomaa [10]. Let Σ be a finite set of symbols, called *letters*; we call Σ an *alphabet*. Then Σ^* is the set of all finite sequences of letters from Σ, which are called *words*. The empty word ε is the empty sequence of letters. We denote by Σ^+ the set of non-empty words over Σ, i.e., $\Sigma^+ = \Sigma^* - \{\varepsilon\}$. The *length* of a word $w = w_1 w_2 \cdots w_n \in \Sigma^*$, where $w_i \in \Sigma$, is n, and is denoted by $|w|$.

A word $x \in \Sigma^*$ is a *prefix* of a word $y \in \Sigma^*$ if there exists $w \in \Sigma^*$ such that $y = xw$. Similarly, x is a *suffix* of y if there exists $u \in \Sigma^*$ such that $y = ux$. If $x \in \Sigma^*$, then $\mathrm{pref}(x)$ (resp., $\mathrm{suff}(x)$) is the set of all prefixes (resp., suffixes) of x. We also use the notation $\mathrm{first}(x)$ and $\mathrm{last}(x)$ to denote the first and last letter of a non-empty word. That is, if $x \in \Sigma^+$ and $x = x_1 x_2$ where $x_1 \in \Sigma$ and $x_2 \in \Sigma^*$, then $\mathrm{first}(x) = x_1$. Similarly, if $x = y_1 y_2$ where $y_1 \in \Sigma^*$ and $y_2 \in \Sigma$, then $\mathrm{last}(x) = y_2$.

A *language* L is any subset of Σ^*. Given languages $L_1, L_2 \subseteq \Sigma^*$, their concatenation is defined by $L_1 L_2 = \{xy \ : \ x \in L_1, y \in L_2\}$. Given an alphabet Σ,

we use the notation Σ^k to denote the set of all words in Σ^* of length k, while $\Sigma^{\geq k}$ (resp., $\Sigma^{\leq k}$) denotes the set of all words in Σ^* of length k or greater (resp., length k or less).

A *deterministic finite automaton* (DFA) is a five-tuple $M = (Q, \Sigma, \delta, q_0, F)$ where Q is the finite set of states, Σ is the alphabet, $\delta : Q \times \Sigma \to Q$ is the transition function, $q_0 \in Q$ is the start state, and $F \subseteq Q$ is the set of final states. We extend δ to $Q \times \Sigma^*$ in the usual way: $\delta(q, \varepsilon) = q$ for all $q \in Q$, while $\delta(q, wa) = \delta(\delta(q, w), a)$ for all $q \in Q$, $w \in \Sigma^*$ and $a \in \Sigma$. A word $w \in \Sigma^*$ is accepted by M if $\delta(q_0, w) \in F$. The *language accepted* by M, denoted $L(M)$, is the set of all words accepted by M, i.e., $L(M) = \{w \in \Sigma^* \ : \ \delta(q_0, w) \in F\}$. A language is called *regular* if it is accepted by some DFA. If L is a regular language, the *state complexity* of L, denoted by $\mathrm{sc}(L)$, is the minimum number of states in any DFA which accepts L.

We assume the reader is familiar with the classes of context-free and recursive languages. A language is a *context-free* if it is generated by a context-free grammar. A language is *recursive* if it is accepted by a Turing machine which halts on all inputs. The classes of regular, context-free and recursive languages form a strict hierarchy of inclusions.

2.1 Template-Guided Recombination

We now give the formal definition of TGR, which was proposed by Prescott *et al.* [9] and first studied as a formal operation by Daley and McQuillan [4]. If $n_1, n_2 \geq 1$ and $x, y, z, t \in \Sigma^*$ are words, we denote by $(x, y) \vdash_{t,n_1,n_2} z$ the fact that we can write

$$x = u_1 \alpha \beta v_1 \tag{1}$$
$$y = v_2 \beta \gamma u_2 \tag{2}$$
$$z = u_1 \alpha \beta \gamma u_2 \tag{3}$$
$$t = \alpha \beta \gamma \tag{4}$$

with $\alpha, \beta, \gamma, u_1, u_2, v_1, v_2 \in \Sigma^*$, $|\alpha|, |\gamma| \geq n_1$ and $|\beta| = n_2$. If n_1, n_2 are understood, then we denote the relation $\vdash_{t,n_1,n_2}$ by $\vdash_t$. The word t is called the *template*.

Intuitively, x and y are the DNA strands which are to be recombined using the template t. The regions v_1 and v_2 represent the internal eliminated sequences (IESs) which do not form part of the final rearranged sequence, and β, which has a minimum length restriction, represents the pointer sequences in the ciliate DNA. Note that in the definition of $(x, y) \vdash_t z$, the words x and y are separate DNA sequences and so TGR is an inter-molecular model for ciliate DNA recombination. Recently, however, an intra-molecular TGR has been considered as well [3].

If $T, L \subseteq \Sigma^*$ are languages, then $\Cap_{T,n_1,n_2}(L)$ is defined by

$$\Cap_{T,n_1,n_2}(L) = \{z \ : \ \exists x, y \in L, t \in T \text{ such that } (x, y) \vdash_{t,n_1,n_2} z\}.$$

Again, we use the notation $\Cap_T(L)$ if n_1, n_2 are understood or unimportant. The language T is the *set of templates*.

We require the following simple observation about TGR:

Observation 1. *If $x, y, z, t \in \Sigma^*$ such that $(x, y) \vdash_{t,n_1,n_2} z$, then $|z| - (|x| + |y|) = -n_2 - (|v_1| + |v_2|)$, where v_1, v_2 are as in (1)–(4).*

We now come to the definition of equivalence for sets of templates. Let $n_1, n_2 \geq 1$. For $T_1, T_2 \subseteq \Sigma^*$, we say that T_1 and T_2 are (n_1, n_2)-*equivalent*, denoted by $T_1 \equiv_{n_1,n_2} T_2$, if $\Cap_{T_1}(L) = \Cap_{T_2}(L)$ for all $L \subseteq \Sigma^*$. By $T_1 \sqsubseteq_{n_1,n_2} T_2$, we mean $\Cap_{T_1}(L) \subseteq \Cap_{T_2}(L)$ for all languages $L \subseteq \Sigma^*$. Note that $T_1 \equiv_{n_1,n_2} T_2$ if and only if $T_1 \sqsubseteq_{n_1,n_2} T_2$ and $T_2 \sqsubseteq_{n_1,n_2} T_1$ hold. We also note that $\equiv_{n_1,n_2}$ is an equivalence relation.

We consider the relationships between $\equiv_{n_1,n_2}$ and $\equiv_{n_1',n_2'}$ for different values of n_1, n_2, n_1', n_2'. We can show that these relations are incomparable.

Theorem 2. *Let Σ be an alphabet with size at least two and $n_1, n_2 \geq 1$. The relations $\equiv_{n_1,n_2}$ and $\equiv_{n_1,n_2+1}$ (resp., $\equiv_{n_1,n_2}$ and $\equiv_{n_1+1,n_2}$) are incomparable.*

3 Language Theoretic Characterization

We can now give our main result, a language-theoretic characterization of equivalence of sets of templates. Let (C1) be the following condition:

$$\begin{aligned} &\forall t, t_1, t_2 \in \Sigma^* \text{ with } |t| = 2n_1 + n_2, \\ &\qquad \text{if } t_1 t t_2 \in T_1 \text{ then } \exists t_1' \in \text{suff}(t_1), t_2' \in \text{pref}(t_2)(t_1' t t_2' \in T_2). \end{aligned} \tag{C1}$$

Condition (C1) is illustrated in Figure 1: for every subword t of length $2n_1 + n_2$ in a template in T_1, there must be an extension of t in T_2 which agrees with the template in T_1 on the subwords flanking t.

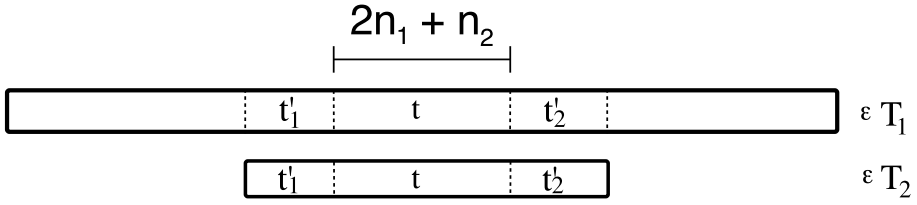

Fig. 1. Illustration of condition (C1).

Our main result uses condition (C1) to characterize equivalence of sets of templates:

Theorem 3. *Let Σ be an alphabet with $|\Sigma| \geq 3$, $n_1, n_2 \geq 1$ and $T_1, T_2 \subseteq \Sigma^*$. The condition (C1) holds if and only if $T_1 \sqsubseteq_{n_1,n_2} T_2$.*

Proof. ($\Rightarrow$): Suppose that (C1) holds. Let L be an arbitrary language and let $x \in \Cap_{T_1}(L)$. Then there exist $y, z \in L, t \in T_1$ such that $(y, z) \vdash_t x$. Write x, y, z, t as

$$\begin{aligned} y &= u_1\alpha\beta v_1, \\ z &= v_2\beta\gamma u_2, \\ x &= u_1\alpha\beta\gamma u_2, \\ t &= \alpha\beta\gamma, \end{aligned}$$

where $|\alpha|, |\gamma| \geq n_1$, $|\beta| = n_2$ and $u_1, u_2, v_1, v_2 \in \Sigma^*$. Now write $\alpha = \alpha_1\alpha_2$ and $\gamma = \gamma_1\gamma_2$ where $|\alpha_2| = n_1$ and $|\gamma_1| = n_1$. Thus, $\alpha_2\beta\gamma_1$ is a subword of t of length $2n_1 + n_2$. By (C1), let $\alpha_1' \in \text{suff}(\alpha_1)$ and $\gamma_2' \in \text{pref}(\gamma_2)$ be chosen so that $t' = \alpha_1'\alpha_2\beta\gamma_1\gamma_2' \in T_2$. Let $\alpha_1 = \alpha_1''\alpha_1'$ and $\gamma_2 = \gamma_2'\gamma_2''$ for appropriate choices of α_1'', γ_2''. Note that

$$\begin{aligned} y &= u_1\alpha_1''(\alpha_1'\alpha_2\beta)v_1, \\ z &= v_2(\beta\gamma_1\gamma_2')\gamma_2''u_2, \\ x &= u_1\alpha_1''\alpha_1'\alpha_2\beta\gamma_1\gamma_2'\gamma_2''u_2. \end{aligned}$$

Thus, $(y, z) \vdash_{t'} x$ and so $x \in \Cap_{T_2}(L)$. We conclude that $T_1 \sqsubseteq_{n_1,n_2} T_2$.

($\Leftarrow$): Suppose for all $L \subseteq \Sigma^*$, we have $\Cap_{T_1}(L) \subseteq \Cap_{T_2}(L)$. Let $t, t_1, t_2 \in \Sigma^*$ with $|t| = 2n_1 + n_2$ and $t_1tt_2 \in T_1$. Let $t_0 = t_1tt_2$. Further, write $t = \alpha\beta\gamma$ where $|\alpha| = |\gamma| = n_1$ and $|\beta| = n_2$. Now, let $X_1, X_2 \in \Sigma$ be letters chosen so that they satisfy

$$\begin{aligned} X_1 &\neq \text{first}(\gamma), & X_2 &\neq \text{last}(\alpha), \\ X_1 &\neq \text{last}(\gamma t_2), & X_2 &\neq \text{first}(t_1\alpha). \end{aligned}$$

Note that this is possible since Σ has at least three letters.

Define the language $L \subseteq \Sigma^*$ as $L = \{t_1\alpha\beta X_1, X_2\beta\gamma t_2\}$. Note that $t_0 \in \Cap_{T_1}(L) \subseteq \Cap_{T_2}(L)$, as $(t_1\alpha\beta X_1, X_2\beta\gamma t_2) \vdash_{t_0} t_0$. Thus, there exist $x, y \in L$ and $t' \in T_2$ such that $(x, y) \vdash_{t'} t_0$. There are three cases, according to the choices for x, y.

(a) $x = t_1\alpha\beta X_1, y = X_2\beta\gamma t_2$. Thus, we must be able to write

$$\begin{aligned} x = t_1\alpha\beta X_1 &= u_1\alpha'\beta' v_1, \\ y = X_2\beta\gamma t_2 &= v_2\beta'\gamma' u_2, \\ t_0 = t_1\alpha\beta\gamma t_2 &= u_1\alpha'\beta'\gamma' u_2, \\ t' &= \alpha'\beta'\gamma', \end{aligned}$$

where $|\alpha'|, |\gamma'| \geq n_1$, $|\beta'| = n_2$. Note that by Observation 1, $|t_1\alpha\beta\gamma t_2| = |t_1\alpha\beta X_1| + |X_2\beta\gamma t_2| - n_2 - |v_1| - |v_2|$. Simplifying, we get that $|v_1| + |v_2| = 2$. We claim that $|v_1| = |v_2| = 1$. If not, then $|v_1| = 2$ and $|v_2| = 0$ or $|v_1| = 0$ and $|v_2| = 2$. We prove that the first case produces a contradiction; the second case is symmetrical.

If $|v_1| = 2$ and $|v_2| = 0$, then the equality $t_1\alpha\beta X_1 = u_1\alpha'\beta' v_1$ implies that $|u_1\alpha'\beta'| = |t_1\alpha\beta| - 1$ and (as $|\beta| = |\beta'| = n_2$) $|t_1\alpha| - 1 = |u_1\alpha'|$. Further $X_2\beta\gamma t_2 = \beta'\gamma' u_2$. Consider now that

$$\begin{aligned} t_1\alpha\beta\gamma t_2 &= u_1\alpha'\beta'\gamma' u_2 \\ &= u_1\alpha' X_2\beta\gamma t_2 \end{aligned}$$

In this case, as $|t_1\alpha| - 1 = |u_1\alpha'|$, we have that $X_2 = \text{last}(\alpha)$, a contradiction to our choice of X_2. (The case $|v_1| = 0$ and $|v_2| = 2$ produces the contradiction that $X_1 = \text{first}(\gamma)$.)
Therefore, $|v_1| = |v_2| = 1$. Thus, $v_1 = X_1, v_2 = X_2$ and we get that $t_1\alpha\beta = u_1\alpha'\beta'$ and $\beta\gamma t_2 = \beta'\gamma' u_2$. We immediately conclude that $\beta = \beta'$ as both have length n_2. As $|\alpha'| \geq n_1 = |\alpha|$, the equality $t_1\alpha\beta = u_1\alpha'\beta'$ implies that there exists $t_1' \in \text{suff}(t_1)$ such that $\alpha' = t_1'\alpha$. Similarly, $\gamma' = \gamma t_2'$ for some $t_2' \in \text{pref}(t_2)$ by the equality $\beta\gamma t_2 = \beta'\gamma' u_2$. Finally, as $t' = \alpha'\beta'\gamma' = t_1'\alpha\beta\gamma t_2'$ and $t' \in T_2$, we note that condition (C1) holds, as required.

(b) $x = X_2\beta\gamma t_2$. Then regardless of the choice of $y \in L$, we have that

$$x = X_2\beta\gamma t_2 = u_1\alpha'\beta' v_1 \tag{5}$$
$$y = v_2\beta'\gamma' u_2 \tag{6}$$
$$t_0 = t_1\alpha\beta\gamma t_2 = u_1\alpha'\beta'\gamma' u_2 \tag{7}$$

Thus, equating (5) and (7), we get that $X_2 = \text{first}(t_1\alpha)$, a contradiction.

(c) $y = t_1\alpha\beta X_1$. This is similar to case (b); we ultimately arrive at the contradiction $X_1 = \text{last}(\gamma t_2)$.

We conclude that in all applicable cases, condition (C1) holds. □

Example 1. Consider the following example with $n_1 = n_2 = 1$:

$$T_1 = \{baaab, caaac\},$$
$$T_2 = \{baaab, caa, aac\}.$$

Note that condition (C1) does *not* hold: for $t = aaa$, $t_1 = t_2 = c$, there is no prefix t_2' of t_2 and suffix t_1' of t_1 such that $t_1' aaa t_2'$ is in T_2. Verifying Theorem 3, we note that $caaac \in \Cap_{T_1}(\{aac, caa\})$, but the same word is not in $\Cap_{T_2}(\{aac, caa\})$.

This example shows that condition (C1) cannot be replaced with the following more simple condition:

$$\forall t \in \text{sub}_{2n_1+n_2}(T_1), \exists t_1', t_2' \in \Sigma^*(t_1' t t_2' \in T_2). \tag{8}$$

(here $\text{sub}_m(L)$ is the set of all subwords of length m in L), since (8) *does* hold for the above sets T_1 and T_2. Intuitively, (8) is not an adequate formulation since it does not enforce that the chosen words t_1', t_2' agree with the regions surrounding the occurrence of t as a subword of length $2n_1 + n_2$ in a template in T_1.

We note that condition (C1) in Theorem 3 does not place any restrictions on templates in T_1 of length less than $2n_1+n_2$. Further, the extensions constructed (i.e., $t'_1tt'_2$ in (C1)) also have length at least $2n_1+n_2$. Thus, there is no restriction on templates less than this critical length $2n_1+n_2$. In other words, if $T_1 \equiv_{n_1,n_2} T_2$, then $T_1 \cap \Sigma^{\leq 2n_1+n_2-1}$ and $T_2 \cap \Sigma^{\leq 2n_1+n_2-1}$ can be modified completely arbitrarily and equivalence will still hold.

Finally, we do not know if the condition $|\Sigma| \geq 3$ in Theorem 3 can be improved to $|\Sigma| \geq 2$. However, the case of $|\Sigma| = 1$ is, as would be expected, trivial. In the case of a unary alphabet, we can replace condition (C1) by the following simpler condition:

$$\forall t \in T_1, |t| \geq 2n_1+n_2,\ \exists t' \in T_2, 2n_1+n_2 \leq |t'| \leq |t|.$$

We omit the proof.

4 Decidability Results

We now turn to employing Theorem 3 to demonstrate that we can determine algorithmically whether two sets of templates are equivalent. We first demonstrate that we can do so if the two sets of templates are regular. To establish this, we show that if T_1 and T_2 do not satisfy (C1), a bound on the length of a template in T_1 demonstrating this fact can be given:

Lemma 1. *Let $T_1, T_2 \subseteq \Sigma^*$ be regular sets of templates, with $sc(T_i) = m_i$ for $i = 1, 2$. If (C1) does not hold, then there exists $t \in T_1$ with $|t| \leq m_1 2^{m_2}+2n_1+n_2$ which witnesses this fact.*

Proof. Let $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$ be DFAs with $|Q_i| = m_i$ and $L(M_i) = T_i$ for $i = 1, 2$.

The proof is by contradiction: Assume that (C1) does not hold. Let $t \in T_1$ be the shortest template that witnesses the fact that (C1) does not hold. Suppose that t has length strictly greater than $m_1 2^{m_2} + 2n_1 + n_2$. As (C1) does not hold, there exists a decomposition of t as $t = t_1 t' t_2$ such that $|t'| = 2n_1 + n_2$, and for all pairs (t'_1, t'_2) where $t'_1 \in \text{suff}(t_1)$ and $t'_2 \in \text{pref}(t_2)$, $t'_1 t t'_2 \notin T_2$.

By the length of t, we must have that either $|t_1| > m_1 2^{m_2-1}$ or $|t_2| > m_1 2^{m_2-1}$. Assume first that $|t_1| > m_1 2^{m_2-1}$. Let $k = |t_1|$ and $t_1 = \eta_1 \eta_2 \cdots \eta_k$ where $\eta_i \in \Sigma$ for all $1 \leq i \leq k$.

For all $1 \leq j \leq k$, let $\Pi_j \subseteq Q_2$ be the set of states

$$\Pi_j = \{\delta_2(q_2, s)\ :\ s \in \text{suff}(\eta_1 \cdots \eta_j)\}.$$

Note that

(a) $q_2 \in \Pi_j$ for all $1 \leq j \leq k$, since $\varepsilon \in \text{suff}(\eta_1 \cdots \eta_j)$.
(b) If $q \in \Pi_j$ and $t'_2 \in \text{pref}(t_2)$, then $\delta_2(q, \eta_{j+1} \cdots \eta_k t' t'_2) \notin F_2$; if this state were in F_2, then the subtemplate $\eta_i \cdots \eta_k t' t'_2 \in T_2$ for some i with $1 \leq i \leq j+1$ (exactly the index i such that $\delta(q_2, \eta_i \cdots \eta_j) = q \in \Pi_j$).

By (a), there are at most 2^{m_2-1} possibilities for Π_j. Then considering all of the pairs $(\Pi_i, \delta_1(q_1, \eta_1 \cdots \eta_i))$ for all $1 \leq i \leq k$, as $k > m_1 2^{m_2-1}$, there must exist $1 \leq j < j' \leq k$ such that $(\Pi_j, \delta_1(q_1, \eta_1 \cdots \eta_j)) = (\Pi_{j'}, \delta_1(q_1, \eta_1 \cdots \eta_{j'}))$.

Claim. The template $t_0 = \eta_1\eta_2 \cdots \eta_j\eta_{j'+1}\eta_{j'+2} \cdots \eta_k t' t_2$ witnesses that (C1) does not hold.

Proof. First, $t_0 \in T_1$. To see this, note that $\delta_1(q_1, \eta_1 \cdots \eta_j) = \delta_1(q_1, \eta_1 \cdots \eta_{j'})$ by choice of j, j', and so substituting the prefix $\eta_1 \cdots \eta_j$ for $\eta_1 \cdots \eta_{j'}$ does not affect the finality of M_1 after reading the entire template, and t_0 is accepted by M_1.

Next, for each suffix t'' of $\eta_1 \cdots \eta_j\eta_{j'+1} \cdots \eta_k$ and each prefix t_2' of t_2 we must have that $t''t't_2' \notin T_2$. For the suffixes of $\eta_{j'+1} \cdots \eta_k$ (and any prefix of t_2), this holds since they are also suffixes of t_1. Consider then a suffix of the form $\eta_i \cdots \eta_j\eta_{j'+1} \cdots \eta_k$ for some $1 \leq i \leq j$. Note that $\delta_2(q_2, \eta_i \cdots \eta_j) \in \Pi_j = \Pi_{j'}$. Thus, there exists a suffix $\eta_r \cdots \eta_{j'}$ of $\eta_1 \cdots \eta_{j'}$ such that $\delta_2(q_2, \eta_i \cdots \eta_j) = \delta_2(q_2, \eta_r \cdots \eta_{j'})$. By (b) above, for all $t_2' \in \text{pref}(t_2)$, $\delta(q_2, \eta_i \cdots \eta_j\eta_{j'+1} \cdots \eta_k t' t_2') = \delta(q_2, \eta_r \cdots \eta_{j'}\eta_{j'+1} \cdots \eta_k t' t_2') \notin F_2$ and thus, $\eta_i \cdots \eta_j\eta_{j'+1} \cdots \eta_k t' t_2' \notin T_2$ for any prefix t_2' of t_2, as required. □

Now, as $j < j'$, we have that t_0 is shorter than t, contrary to our assumption that t was the shortest template in T_1 such that (C1) does not hold. The case where $|t_2| > m_1 2^{m_2-1}$ is similar. Thus, we must have that $|t| \leq m_1 2^{m_2} + 2n_1 + n_2$. □

Corollary 1. *Let $n_1, n_2 \geq 1$ and $T_1, T_2 \subseteq \Sigma^*$ ($|\Sigma| \geq 3$) be effective regular sets of templates. Then it is decidable whether $T_1 \equiv_{n_1,n_2} T_2$.*

Proof. We can assume without loss of generality that $T_1, T_2 \subseteq \Sigma^{\geq 2n_1+n_2}$, as we have observed that templates of length less than this critical length do not affect equivalence.

By Theorem 3, $T_1 \equiv_{n_1,n_2} T_2$ if and only if (C1) holds twice, with T_1 and T_2 in both roles. To test (C1), it suffices to test all words up to the length given by Lemma 1. □

Note that Corollary 1 is not an efficient algorithm: it requires checking an exponential number of templates up to a bound which is itself exponential in the size of the minimal DFA for T_1.

We note the following alternative proof for Corollary 1 which does not use Lemma 1, suggested to us by an anonymous referee. Let $t \in \Sigma^*$ and $T \subseteq \Sigma^*$ be arbitrary, and let $\# \notin \Sigma$. Define $T \ddagger t = \{t_1 \# t_2 \ : \ t_1 t t_2 \in T\}$. Note that if t is not a subword of t', then t' does not contribute anything to $T \ddagger t$. It is not difficult to demonstrate that $T \ddagger t$ is regular for all regular sets of templates T and all $t \in \Sigma^*$. We then note that

$$T_1 \sqsubseteq_{n_1,n_2} T_2 \iff \forall t \in \Sigma^{2n_1+n_2}, T_1 \ddagger t \subseteq \Sigma^*(T_2 \ddagger t)\Sigma^*.$$

That is, $T_1 \sqsubseteq_{n_1,n_2} T_2$ if and only if every word in $T_1 \ddagger t$ has a subword in $T_2 \ddagger t$. This subword must necessarily have an occurrence of $\#$, which has effectively

replaced t, and so we capture (C1) exactly. Therefore, the process of testing the above condition for all words of length $2n_1 + n_2$ gives an alternate method of deciding whether $T_1 \sqsubseteq_{n_1,n_2} T_2$.

We can now give a somewhat surprising positive decidability result for recursive sets of templates. In particular, we can establish a universality equivalence result:

Theorem 4. *Let $n_1, n_2 \geq 1$ and Σ be an alphabet of size at least three. Given an effectively recursive set of templates $T \subseteq \Sigma^*$, we can determine whether $T \equiv_{n_1,n_2} \Sigma^*$.*

However, we also have the following result, which demonstrates that there is at least one regular set of templates such that determining equivalence for context-free sets of templates is undecidable:

Theorem 5. *Let Δ be an alphabet of size at least three and $n_1, n_2 \geq 1$. There exists a fixed regular set of templates $T_0 \subseteq \Delta^*$ such that the following problem is undecidable: Given a context-free set of templates $T \subseteq \Delta^*$, is $T \equiv_{n_1,n_2} T_0$?*

5 Conclusions

In this paper, we have considered equivalence of sets of templates. With a natural condition on extending subwords of the critical length $2n_1 + n_2$ in one set of templates to a template in the equivalent set, we have exactly characterized the equivalence of two sets of templates for alphabets of size three or more, which is sufficient for modelling biological processes. It is open whether the construction can be reduced to an alphabet of size two.

Using this characterization, we have shown that it is decidable whether two regular sets of templates are equivalent. This uses a result which establishes that if two regular sets of templates are not equivalent, a witness can be found within some finite bound. We have also established two other decidability results. First, deciding equivalence to the set of all possible templates is easier than might be expected: we can determine such an equivalence for recursive sets of templates. However, there exists a fixed regular set of templates T_0 such that it is undecidable whether a given context-free set of templates is equivalent to T_0.

We mention the problem of equivalence for iterated TGR, which has been defined as a formal operation by Daley and McQuillan [4]. Iterated TGR serves as a more realistic biological model of DNA rearrangement in ciliates. It is not difficult to show that if $T_1 \equiv_{n_1,n_2} T_2$, then the iterated TGR operations using T_1 and T_2 are also equivalent. Thus, equivalence of two sets of templates implies the equality of the corresponding iterated TGR operations using T_1 and T_2. However, the converse, i.e., whether equivalence of templates in iterated TGR implies equivalence for non-iterated TGR, is open and a topic for future research.

Acknowledgments

We thank the referees of DNA 13 for their helpful comments, and in particular, the suggested alternative proof of Corollary 1.

References

1. Angeleska A., Jonoska, N., Saito, M., Landweber, L. RNA-Guided DNA Assembly. Journal of Theoretical Biology (to appear), 2007. Abstract appears in *DNA 13: Preliminary Proceedings* (2007), M. Garzon and H. Yan, Eds., p. 364
2. Cavalcanti, A., Landweber, L. Insights into a biological computer: detangling scrambled genes in ciliates. In *Nanotechnology: Science and Computation*, J. Chen, N. Jonoska, and G. Rozenberg, Eds. Springer-Verlag, 2006, pp. 349–360
3. Daley, M., Domaratzki, M., Morris, A. Intra-molecular template-guided recombination. International Journal of Foundations of Computer Science (to appear), 2007. Preliminary technical report available at `http://www.cs.acadiau.ca/research/technicalReports`
4. Daley, M., McQuillan, I. Template-guided DNA recombination. Theoretical Computer Science **330** (2005), 237–250
5. Daley, M., McQuillan, I. On computational properties of template-guided DNA recombination in ciliates. In *DNA Computing* (2006), A. Carbone and N. Pierce, Eds., vol. 3892 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 27–37
6. Daley, M., McQuillan, I. Useful templates and iterated template-guided DNA recombination in ciliates. Theory of Computing Systems **39** (2006), 619–633
7. Ehrenfeucht, A., Harju, T., Petre, I., Prescott, D., Rozenberg, G. *Computation in Living Cells: Gene Assembly in Ciliates*. Springer-Verlag, 2004
8. McQuillan, I., Salomaa, K., Daley, M. Iterated TGR languages: Membership problem and effective closure properties. In *Computing and Combinatorics* (2006), D. Chen and D. Lee, Eds., vol. 4112 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 94–103
9. Prescott, D., Ehrenfeucht, A., Rozenberg, G. Template-guided recombination for IES elimination and unscrambling of genes in stichotrichous ciliates. Journal of Theoretical Biology **222** (2003), 323–330
10. Rozenberg, G., Salomaa, A., Eds. *Handbook of Formal Languages*. Springer-Verlag, 1997
11. Vijayan, V., Nowacki, M., Zhou, Y., Doak, T., Landweber, L. Programming a Ciliate Computer: Template-Guided IN Vivo DNA Rearrangements in *Oxytricha*. In *DNA 13: Preliminary Proceedings* (2007), M. Garzon and H. Yan, Eds., p. 172