Small fast universal Turing machines

Turlough Neary[1] and Damien Woods[2]

1. TASS research group, Department of Computer Science,
NUI Maynooth, Ireland.
tneary@cs.may.ie
2. Boole Center for Research in Informatics, School of Mathematics,
University College Cork, Ireland.
d.woods@bcri.ucc.ie

Communicated by: J. Paul Gibson
Technical report: NUIM-CS-2005-TR-11

Key words: small universal Turing machine, polynomial time, universality.

Abstract. We present a number of time-efficient small universal Turing machines. We show that there exists deterministic polynomial time universal Turing machines with state-symbol products of $(3, 11)$, $(5, 7)$, $(6, 6)$, $(7, 5)$ and $(8, 4)$. These machines are the smallest known universal Turing machines that simulate TMs in polynomial time.

1 Introduction

Shannon [1] first posed the question of finding the smallest possible universal Turing machine (UTM). Initially small UTMs were constructed that directly simulated Turing machines (TMs) [2, 3]. Subsequently the technique of indirect simulation via other universal models was successfully applied. In the early 1960s Minsky used 2-tag systems to create a 7-state, 4-symbol machine [4]. Minsky's technique was more recently used by Rogozhin et al to create the smallest known UTMs.

Let UTM(m, n) be the class of deterministic UTMs with m states and n symbols. Rogozhin [5] constructed UTMs in the classes UTM$(24, 2)$ UTM$(10, 3)$, UTM$(7, 4)$, UTM$(5, 5)$, UTM$(4, 6)$, UTM$(3, 10)$ and UTM$(2, 18)$, Kudlek and Rogozhin [6] constructed a machine in UTM$(3, 9)$, and Baiocchi [7] constructed UTMs in UTM$(19, 2)$ and UTM$(7, 4)$. In terms of the number of transition rules (TRs), Baiocchi's 4-symbol UTM is the smallest in the class UTM$(7, 4)$. Due to a unary encoding of the TM tape contents 2-tag systems are exponentially slow simulators of TMs. Hence the simulations of Minsky, Rogozhin, Kudlek and Baiocchi all suffer from an exponential time complexity overhead. Fig. 1 is a state-symbol plot, here we see that these machines induce a curve which we call the exponential time curve. The halting problem has been proved decidable for all deterministic TMs in the classes TM$(2, 2)$ [8, 9], TM$(3, 2)$ [10], TM$(2, 3)$ (Pavlotskaya unpublished), TM$(1, n)$ [11] and TM$(n, 1)$ (trivial) for $n \geq 1$. These results induce the non-universal curve in Fig. 1.

Our main result states that there exists deterministic polynomial time UTMs in the classes UTM$(3, 11)$, UTM$(5, 7)$, UTM$(6, 6)$, UTM$(7, 5)$ and UTM$(8, 4)$. Fig. 1 illustrates the polynomial time curve that is induced by our result. It follows immediately

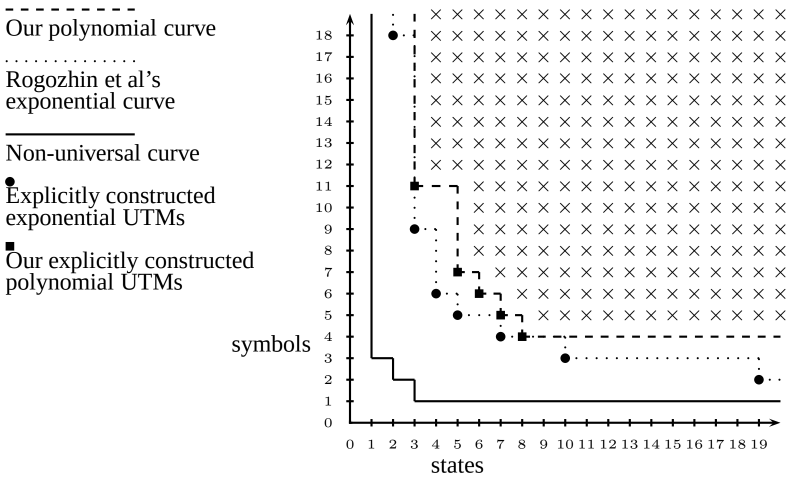

Fig. 1. State-symbol plot of small UTMs. The plot shows the polynomial curve induced by our machines. Rogozhin et al's exponential time curve, and the current non-universal TM curve. The $\times$ symbols and the polynomial curve represent the main contribution of this paper; there are polynomial time UTMs with these state-symbol values

that there are efficient polynomial time UTMs with state-symbol products indicated by the crosses ($\times$) in Fig. 1. It is interesting to note that in some places our polynomial time curve actually intersects the exponential time curve.

Before our work the most recent small polynomial time UTM was constructed by Watanabe [3] in 1961 and is in the class UTM$(8, 5)$. Subsequent efforts to construct smaller UTMs have used the (exponentially slow) technique of simulation via 2-tag systems. Our results offer a significant improvement over Watanabe's 1961 machine; our machines are significantly smaller and represent a new algorithm for small UTMs.

In Section 2 we give an overview of our simulation algorithm and some definitions used to encode input to our UTMs. In Section 3 we give a machine in the class UTM$(3, 11)$. We explain its input encoding and operation in some detail. Section 4 contains a proof of correctness which proves that this UTM simulates TMs in polynomial time. In the remaining sections our algorithm is extended to UTMs with a number of other state-symbol products and finally a conclusion is given.

2 Preliminaries

Boas [12] discusses how difficult it is to define *simulation* as a mathematical object and still remain sufficiently general. Rogozhin [5] gives formal definitions of simulation between TMs and of UTM. In both of these definitions the encoding and decoding functions are recursive. Our UTMs satisfy Rogozhin's definitions and also simulate deterministic TMs in polynomial time. Rogozhin et al's UTMs simulate deterministic TMs in exponential time. It is not known if tag systems can simulate TMs without using a unary encoding hence it is not known if Rogozhin et al's UTMs can simulate TMs in polynomial time.

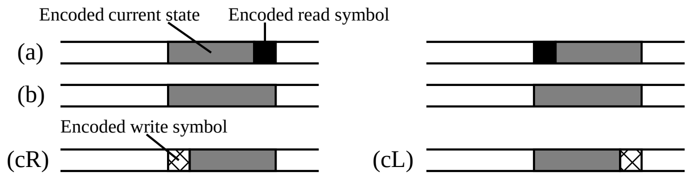

Fig. 2. Right and left shifting transition rule simulations. The encoded current state marks the location of M's simulated tape head. (a) Encoded configurations before beginning each TR simulation. (b) Intermediate configurations immediately after the encoded read symbol and encoded current state have been read. (cR) Configuration immediately after the simulated right shift. (cL) Configuration immediately after the simulated left shift.

Before going into technical details we describe the main difference between our UTMs and previous small polynomial time UTMs and why this difference is significant. We then introduce some general encodings that each of our five machines adhere to. We also give an overview of our simulation algorithm. Each UTM uses a variation on this algorithm.

2.1 General form of our universal machines

In order to distinguish the current state q_x of a simulated TM M earlier small UTMs [2, 3] maintained a list of all states with a marker at q_x. A change in M's current state was simulated by moving the marker to another location in the list of states. The most significant difference between these earlier UTMs and our algorithm is that we store the encoded current state of M on M's simulated tape at the location of M's tape head. Thus the encoded current state also records the current location of M's tape head during the simulation. This point is illustrated in Fig. 2.

The problem of constructing a UTM can be broken into the following basic steps. The UTM must (1) read the encoded current state and (2) read the encoded read symbol. Next the UTM must (3) print the encoded write symbol, (4) move the simulated tape head and (5) establish the new encoded current state. Due to the location of the encoded current state and the special encodings we use for our UTMs the sets {(1), (2)} and {(3), (4), (5)} each become a single process. Steps (1) and (2) are combined such that a single set of transition rules read both the encoded current state and the encoded read symbol. Steps (3), (4) and (5) have been similarly combined. Combining these steps has reduced the number of transition rules needed by our UTMs.

2.2 TMs

We consider deterministic TMs with a single one-way infinite tape and a single tape head [13]. A TM is a tuple $M = (Q, \Sigma, B, f, q_1, H)$ (adapted from [13]). Q and Σ are the finite sets of states and tape symbols respectively. $B \in \Sigma$ is the blank symbol. $q_1 \in Q$ is the start state and $H \subseteq Q$ is the set of halt states. The transition function $f : Q \times \Sigma \rightarrow Q \times \Sigma \times \{L, R\}$ is total for $q \in Q$ if $q \notin H$. If $q \in H$ the function f is partial, that is f is undefined on at least one element of $q \times \Sigma$. We write f as a list of

TRs. Each TR is a quintuple $t = (q_x, \sigma_1, \sigma_2, D, q_y)$, with *initial* state q_x, read symbol σ_1, write symbol σ_2, shift direction D and *next* state q_y.

Throughout the paper U denotes a UTM and for some $m, n \in \mathbb{N}$, $U_{m,n}$ denotes our UTM in class UTM(m, n). We let M always denote a TM that is to be simulated by some U. The encoding of M as a word is denoted $\widehat{M}$. Analogously the encodings of state q and tape symbol σ are denoted $\widehat{q}$ and $\widehat{\sigma}$ respectively. For convenience we often call the word $\widehat{q}$ a *state* of $\widehat{M}$. We let $\mathbb{N}$ denote the set of nonnegative integers. In regular expressions $\cup$, *, ϵ and parentheses have their usual meanings [13].

2.3 Input encodings for UTMs

Without loss of generality, any simulated TM M has the following restrictions: (i) M's tape alphabet is $\Sigma = \{0, 1\}$, (ii) for all $q_i \in Q$, i satisfies $1 \leq i \leq |Q|$, (iii) f is total, (iv) M's start state is q_1, (v) M has exactly one halt state $q_{|Q|}$ and its transition rules are of the form $(q_{|Q|}, 0, 0, L, q_{|Q|})$ and $(q_{|Q|}, 1, 1, L, q_{|Q|})$. Point (v) is a well-known halting technique where the tape head is placed at the beginning of the output. The following definitions encode M.

Each of our five UTMs has the symbols $\overleftarrow{a}$, $\overleftarrow{b}$ and λ as part of its tape alphabet. The symbol λ is typically used as a marker symbol while $\overleftarrow{a}$ and $\overleftarrow{b}$ are usually used to encode M's tape symbols as follows.

Definition 1 (Encoding of M's tape symbols). *The binary tape symbols 0 and 1 of M are encoded as the words $\widehat{0} = \overleftarrow{a}\overleftarrow{a}$ and $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$.*

Definition 2 (Encoding of M's initial configuration). *The encoding of an initial configuration of M is of the form*

$$\widehat{M}\widehat{q_1}\widehat{w}(\overleftarrow{a})^{\omega}$$

where $\widehat{q_1}$ is $\widehat{M}$'s start state, $\widehat{w} \in \{\overleftarrow{a}\overleftarrow{a}, \overleftarrow{b}\overleftarrow{a}\}^*$ is the encoding of input to M given by Definition 1, $(\overleftarrow{a})^{\omega} = \overleftarrow{a}\overleftarrow{a}\overleftarrow{a}\cdots$ and $\widehat{M}$ is the encoding of M:

$$\widehat{M} = \lambda\mathcal{P}(f, q_{|Q|})\lambda\mathcal{P}(f, q_{|Q|-1})\lambda\cdots\lambda\mathcal{P}(f, q_2)\lambda\mathcal{P}(f, q_1)\lambda E \tag{1}$$

where the function $\mathcal{P}$ is defined below in Equation (2)*, and the word $E \in \{\epsilon, e, \overleftarrow{a}, \lambda\overleftarrow{b}\lambda\overleftarrow{a}, \overleftarrow{b}\overleftarrow{b}\overleftarrow{b}\lambda\overleftarrow{a}\}$ specifies the* ending.

The initial position of U's tape head is at the leftmost symbol of $\widehat{q_1}$.

In the previous definition the encoding of M is placed to the left of its encoded input. The initial position of M's simulated tape head is indicated by the word $\widehat{q_1}$ and is immediately to the left of the leftmost encoded input symbol. The remainder of the infinite tape of U contains the blank symbol $\overleftarrow{a}$. The ending E varies over the five UTMs that we present.

The encoding of M's TRs is defined using the function $\mathcal{P}$ that specifies the relative *positions* of encoded TRs for a given state q_i.

$$\mathcal{P}(f, q_i) = \mathcal{E}(t_{i,1})\lambda\mathcal{E}(t_{i,0})\lambda\mathcal{E}(t_{i,0})\lambda\mathcal{E}(t_{i,1})\lambda\mathcal{E}'(f, t_{i,0}) \tag{2}$$

The encoding functions $\mathcal{E}$ and $\mathcal{E}'$ map TRs to words called ETRs. There is a specific pair of $\mathcal{E}$ and $\mathcal{E}'$ functions for each of our five UTMs. Given what we have so far, we need only to give $\mathcal{E}$ and $\mathcal{E}'$ to completely define the input to our UTMs. These functions are given before each UTM.

2.4 UTM algorithm overview

Here we give a brief description of the simulation algorithm. The encoded current state of M is positioned at the simulated tape head location of M. Using a unary indexing method, U locates the next ETR to execute. The next ETR is indexed (pointed to) by the number of $\overleftarrow{b}$ symbols contained in the encoded current state and read symbol. If the encoded current state and read symbol together contain i of the $\overleftarrow{b}$ symbols then there will be $i-1$ of the λ markers between the encoded current state and the next ETR to be executed. To locate the next ETR, U simply neutralises the rightmost λ (i.e. changes λ to some other symbol) for each $\overleftarrow{b}$ in the encoded current state and encoded read symbol, until there is only one $\overleftarrow{b}$ remaining. This indexed ETR is printed over the previous encoded current state and read symbol. This printing completes the execution of the new ETR and establishes the new encoded current state, encoded write symbol and simulated tape head shift. Fig. 2(b) represents the tape contents of U after an ETR of $\widehat{M}$'s has been indexed. Fig. 2(cR) and (cL) represent the two possibilities for U's tape contents after an ETR has been printed. To give more details we decompose the algorithm into four cycles.

Cycle 1 (Index next ETR)
In Cycle 1 U reads the encoded current state and read symbol and neutralises markers to index the next ETR. Initially U's tape head scans to the right and when it reads a $\overleftarrow{b}$ it changes the $\overleftarrow{b}$ to some other symbol. U's tape head then scans left to neutralise a λ marker. This process is repeated until U reads the substring $\overleftarrow{b}\,\overleftarrow{a}$ while scanning right. This signals the end of Cycle 1 and the beginning of Cycle 2.

Cycle 2 (Print ETR)
Cycle 2 copies an ETR to M's simulated tape head location. U scans left and records the next symbol of the ETR to be printed. U then scans right and prints the next symbol of the ETR at a location specified by a marker. The location of this marker is initially set at the end of Cycle 1 and its location is updated after the printing of each symbol of the ETR. This process is repeated until the end of the ETR is detected causing U to enter Cycle 3. The end of the ETR is detected by U encountering the marker or neutralised marker that separates two ETRs.

Cycle 3 (Restore tape)
Cycle 3 restores M's encoded table of behaviour after an ETR has been indexed and printed over the old encoded current state and encoded read symbol. U scans right restoring $\widehat{M}$ to its initial value. This Cycle ends when U encounters the marker which was used in Cycle 2 to specify the position of the next symbol of the ETR to be printed. When U's tape head reads this symbol U enters Cycle 4.

Cycle 4 (Choose read or write symbol)
This cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if U is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if U is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state. On completion of either case Cycle 1 is entered.

3 Construction of $U_{3,11}$

Our first machine is in class UTM$(3, 11)$ and is denoted $U_{3,11}$. As usual let M be a TM that is simulated by $U_{3,11}$.

Definition 3 **($\widehat{M}$'s start state).** *The start state of $\widehat{M}$ is encoded as $\widehat{q_1} = \overleftarrow{a}^{5|Q|}\overleftarrow{b}^{2}$.*

Recall that $\widehat{M}$ is the encoding of M and is defined via the functions $\mathcal{E}$ and $\mathcal{E}'$. The functions $\mathcal{E}$ and $\mathcal{E}'$ map to words over the alphabet of $U_{3,11}$. These functions encode TRs and are given by Equations (3) and (4) respectively. We denote the words defined by $\mathcal{E}$ and $\mathcal{E}'$ with the acronyms ETR and ETR′ respectively.

We use a shorthand notation for TRs. We let $t_{i,\sigma_1} = (q_i, \sigma_1, \sigma_2, D, q_y)$, that is t_{i,σ_1} denotes the unique TR in M with initial state q_i and read symbol σ_1. Also $t^{R,i} = (q_x, \sigma_1, \sigma_2, R, q_i)$ and $t^{L,i} = (q_x, \sigma_1, \sigma_2, L, q_i)$; we write $\exists t^{R,i}$ to mean that there exists a TR which shifts right and has q_i as its next state (there may be zero or more such TRs).

Let $t = (q_x, \sigma_1, \sigma_2, D, q_y)$ be a fixed TR in M, then t is encoded via $\mathcal{P}$ using the function $\mathcal{E}$ on its own or in conjunction with $\mathcal{E}'$ where

$$\mathcal{E}(t) = \begin{cases} e^{a(t)}h^{b(t)} & \text{if } D = R, \sigma_2 = 0, \\ he^{a(t)}h^{b(t)} & \text{if } D = R, \sigma_2 = 1, \\ e^{a(t)-1}h^{b(t)}eee & \text{if } D = L, \sigma_2 = 0, \\ e^{a(t)-1}h^{b(t)}ehe & \text{if } D = L, \sigma_2 = 1, \end{cases} \tag{3}$$

and

$$\mathcal{E}'(f,t) = \begin{cases} e^{a(t^{R,x})-3}h^{b(t^{R,x})+2} & \text{if } \exists t^{R,x}, q_x \neq q_1, \\ e^{5|Q|-3}h^4 & \text{if } q_x = q_1, \\ \epsilon & \text{if } \neg\exists t^{R,x}, q_x \neq q_1, \end{cases} \tag{4}$$

where as before $t^{R,x}$ is any right shifting TR such that $t^{R,x} \vdash t$, the functions $a(\cdot)$ and $b(\cdot)$ are defined by Equations (5) and (6), e and h are tape symbols and ϵ is the empty word.

$$a(t) = 5|Q| + 2 - b(t), \tag{5}$$

$$b(t) = 2 + \sum_{j=1}^{y} g(t,j), \tag{6}$$

where $g(\cdot)$ is given by

$$g(t,j) = \begin{cases} 5 & \text{if } j < y, \\ 3 & \text{if } D = L, j = y, \\ 0 & \text{if } D = R, j = y. \end{cases} \tag{7}$$

Definition 4 **(Encoding of M's current state).** *The encoding of M's current state is of the form $\overleftarrow{a}^{*}\overleftarrow{b}^{2}\overleftarrow{b}^{*}\{\overleftarrow{a} \cup \epsilon\}$ and is of length $5|Q| + 2$.*

The value of E from Definition 2 for $U_{3,11}$ is $E = e$.

ETR	Transition Rule	$t^{R,x}$ for $\mathcal{E}'$	$b(t)$	$a(,t)$
$\mathcal{E}'(f,t_{1,0})$				
$\mathcal{E}(t_{1,1})$	$q_1, 1, 0, R, q_1$		$2+0=2$	15
$\mathcal{E}(t_{1,0})$	$q_1, 0, 1, R, q_2$		$2+5+0=7$	10
$\mathcal{E}(t_{1,0})$	$q_1, 0, 1, R, q_2$		$2+5+0=7$	10
$\mathcal{E}(t_{1,1})$	$q_1, 1, 0, R, q_1$		$2+0=2$	15
$\mathcal{E}'(f,t_{2,0})$	$q_2, 0, 0, L, q_2$	$q_1, 0, 1, R, q_2$	$2+5+0=7$	10
$\mathcal{E}(t_{2,1})$	$q_2, 1, 1, L, q_3$		$2+5+5+3=15$	2
$\mathcal{E}(t_{2,0})$	$q_2, 0, 0, L, q_2$		$2+5+3=10$	7
$\mathcal{E}(t_{2,0})$	$q_2, 0, 0, L, q_2$		$2+5+3=10$	7
$\mathcal{E}(t_{2,1})$	$q_2, 1, 1, L, q_3$		$2+5+5+3=15$	2
$\mathcal{E}'(f,t_{3,0})$	$q_3, 0, 0, L, q_3$	null	null	null
$\mathcal{E}(t_{3,1})$	$q_3, 1, 1, L, q_3$		$2+5+5+3=15$	2
$\mathcal{E}(t_{3,0})$	$q_3, 0, 0, L, q_3$		$2+5+5+3=15$	2
$\mathcal{E}(t_{3,0})$	$q_3, 0, 0, L, q_3$		$2+5+5+3=15$	2
$\mathcal{E}(t_{3,1})$	$q_3, 1, 1, L, q_3$		$2+5+5+3=15$	2

Table 1. Values for the $a(\cdot)$ and $b(\cdot)$ functions for each ETR of $\widehat{M_1}$.

Example 1 (Encoding of M_1). Let TM $M_1 = (\{q_1, q_2, q_3\}, \{0, 1\}, 0, f, q_1, \{q_3\})$ where f is defined by $(q_1, 0, 1, R, q_2)$, $(q_1, 1, 0, R, q_1)$, $(q_2, 0, 0, L, q_2)$, $(q_2, 1, 1, L, q_3)$, $(q_3, 0, 0, L, q_3)$ and $(q_3, 1, 1, L, q_3)$. From Equation (1) M_1 is encoded as:

$$\widehat{M_1} = \lambda\mathcal{P}(f, q_3)\lambda\mathcal{P}(f, q_2)\lambda\mathcal{P}(f, q_1)\lambda e$$

$\widehat{M_1}$'s start state is $\overleftarrow{a}^{15}\overleftarrow{b}^{2}$. Substituting the appropriate values from Equation (2) gives

$$\begin{aligned}\widehat{M_1} =& \lambda\mathcal{E}(t_{3,1})\lambda\mathcal{E}(t_{3,0})\lambda\mathcal{E}(t_{3,0})\lambda\mathcal{E}(t_{3,1})\lambda\mathcal{E}'(f, t_{3,0})\\ &\lambda\mathcal{E}(t_{2,1})\lambda\mathcal{E}(t_{2,0})\lambda\mathcal{E}(t_{2,0})\lambda\mathcal{E}(t_{2,1})\lambda\mathcal{E}'(f, t_{2,0})\\ &\lambda\mathcal{E}(t_{1,1})\lambda\mathcal{E}(t_{1,0})\lambda\mathcal{E}(t_{1,0})\lambda\mathcal{E}(t_{1,1})\lambda\mathcal{E}'(f, t_{1,0})\lambda e\end{aligned}$$

Rewriting this using Equations (3) and (4) and the values given in Table 1 gives the word

$$\begin{aligned}\widehat{M_1} =& \lambda eh^{15}ehe\lambda eh^{15}eee\lambda eh^{15}eee\lambda eh^{15}ehe\lambda\epsilon\lambda eh^{15}ehe\lambda e^{6}h^{10}eee\lambda\\ &e^{6}h^{10}eee\lambda eh^{15}ehe\lambda e^{7}h^{9}\lambda e^{15}h^{2}\lambda he^{10}h^{7}\lambda he^{10}h^{7}\lambda e^{15}h^{2}\lambda e^{12}h^{4}\lambda e\end{aligned} \tag{8}$$

To aid understanding, note that a key property of $\mathcal{P}$ from Equation (2) is that it creates five ETRs in $\widehat{M}$ for each state in M. Hence five ETRs encode two TRs. Our reason for this is because of the algorithm used by our UTMs. When executing an ETR, the algorithm makes use of the *direction* of the *previous* tape head movement of M. The leftmost ETR given by Equation (2) simulates execution of TR $t_{i,1}$ following a simulated left shift. The second ETR from the left simulates execution of TR $t_{i,0}$ following a simulated left shift. The rightmost ETR and the middle ETR are both used to simulate execution of TR $t_{i,0}$ following a simulated right shift. Finally the second ETR from the right simulates execution of TR $t_{i,1}$ following a simulated right shift. For a given state q_i and a given $t^{L,i}$, the function $\mathcal{P}$ places the encoding for the TR with read symbol 0 to the right of the encoding of the TR with read symbol 1. For a given $t^{R,i}$, $\mathcal{P}$ places the encoding of $t_{i,1}$ *between* the ETR and ETR′ that encode $t_{i,0}$.

In our simulation, the number of $\overleftarrow{b}$ symbols in the encoded current state is used as a unary index to locate the *next* ETR to be executed. The functions $a(\cdot)$ and $b(\cdot)$ defined by Equations (5) and (6) give the ratio of e to h symbols in an ETR. The number of h symbols in the ETR being executed defines the number of $\overleftarrow{b}$ symbols in the *next* encoded current state ($\widehat{q_y}$). The word $\mathcal{P}(f, q_y)$ gives the ETRs that encoded the TRs for state q_y. Hence the next ETR to be indexed is a subword of $\mathcal{P}(f, q_y)$ and $b(\cdot)$ is a summation dependant on all encoded states $\widehat{q_j}$ such that $j \leq y$. The function g defined by Equation (7) is used by $b(\cdot)$ to calculate the number of ETRs in each $\widehat{q_j}$. The first value of g corresponds exactly to the number of ETRs given in $\mathcal{P}$ (Equation (2)). The final two values of g define whether the encoded current state the ETR establishes points to the rightmost ETR ($g = 0$) in the list of ETRs for a state, or to the fourth from the right ($g = 3$).

3.1 $\boldsymbol{U_{3,11}}$ and its computation

Definition 5 **($U_{3,11}$).** *The TM* $U_{3,11}$ *is defined as* $U_{3,11} = (\{u_1, u_2, u_3\}, \{\overleftarrow{a}, \overleftarrow{b}, e, h, \overrightarrow{e}, \overrightarrow{h}, \overleftarrow{e}, \overleftarrow{h}, \lambda, \delta, \gamma\}, \overleftarrow{a}, f, u_1, \{u_3\})$ *where* f *is given by the following transition rules.*

$u_1, \overleftarrow{a}, \overleftarrow{e}, R, u_1$	$u_2, \overleftarrow{a}, \gamma, L, u_2$	$u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{e}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_1$	$u_3, \overleftarrow{b}, e, R, u_1$
$u_1, e, \overrightarrow{e}, L, u_1$	$u_2, e, \overleftarrow{e}, R, u_1$	u_3, e, e, R, u_1
$u_1, h, \overrightarrow{h}, L, u_1$	$u_2, h, \overleftarrow{h}, R, u_3$	$u_3, h, \overleftarrow{a}, L, u_1$
$u_1, \overrightarrow{e}, \overleftarrow{e}, R, u_1$	$u_2, \overrightarrow{e}, e, R, u_2$	$u_3, \overrightarrow{e}, \overleftarrow{e}, R, u_3$
$u_1, \overrightarrow{h}, \overleftarrow{h}, R, u_1$	$u_2, \overrightarrow{h}, h, R, u_2$	$u_3, \overrightarrow{h}, \overleftarrow{h}, R, u_3$
$u_1, \overleftarrow{e}, \overrightarrow{e}, L, u_1$	$u_2, \overleftarrow{e}, \overrightarrow{e}, L, u_2$	$u_3, \overleftarrow{e}, \gamma, L, u_2$
$u_1, \overleftarrow{h}, \overrightarrow{h}, L, u_1$	$u_2, \overleftarrow{h}, \overrightarrow{h}, L, u_2$	$u_3, \overleftarrow{h}, \overleftarrow{a}, L, u_3$
$u_1, \lambda, \delta, R, u_1$	$u_2, \lambda, \lambda, R, u_2$	$u_3, \lambda, \delta, R, u_3$
$u_1, \delta, \lambda, L, u_1$	$u_2, \delta, \lambda, L, u_2$	$u_3, \delta,$
$u_1, \gamma, \overleftarrow{a}, L, u_3$	$u_2, \gamma, \overleftarrow{h}, R, u_3$	$u_3, \gamma, \overleftarrow{b}, L, u_3$

We give an example of $U_{3,11}$ simulating a TR of M_1 from Example 1. This simulation is of the first step in M_1's computation for a specific input. The example is broken down into the 4 cycles given in Section 2.4. The current state of $U_{3,11}$ is highlighted in bold font, to the left of $U_{3,11}$'s tape contents. M_1's encoded read and write symbols are also highlighted in bold font. The position of $U_{3,11}$'s tape head is given by an underline. In the sequel we use the term *overlined region*.

Definition 6 **(Overlined region).** *The overlined region exactly spans the encoded current state (has length* $5|Q|+2$*) except on completion of reading an encoded read symbol (has length* $5|Q|+4$*). In the latter case the overlined region exactly spans the encoded current state and read symbol, until the next encoded current state is established.*

Example 2 *($U_{3,11}$'s simulation of TR* $t_{1,1} = (q_1, 1, 0, R, q_1)$ *from TM* M_1*).* The start state of $U_{3,11}$ is u_1 and $U_{3,11}$'s tape head is over the symbol directly to the right of $\widehat{M_1}$ (as in Definition 2). In this example M_1's input is 101 (encoded via $\widehat{0} = \overleftarrow{a}\,\overleftarrow{a}$ and $\widehat{1} = \overleftarrow{b}\,\overleftarrow{a}$). $\widehat{M_1}$ is in start state $\widehat{q_1}$ with encoded read symbol $\widehat{1}$. Thus the initial configuration of U is:

$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\lambda e\,\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}^{14}\,\overleftarrow{b}\,\overleftarrow{b}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 1 (Index next ETR)

$u_1, \overleftarrow{a}, \overleftarrow{e}, R, u_1$	$u_2, \overleftarrow{a}, \gamma, L, u_2$	$u_1, e, \overrightarrow{e}, L, u_1$
$u_1, \overleftarrow{b}, \overleftarrow{e}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_1$	$u_1, h, \overrightarrow{h}, L, u_1$
$u_1, \overrightarrow{e}, \overleftarrow{e}, R, u_1$		$u_1, \overleftarrow{e}, \overrightarrow{e}, L, u_1$
$u_1, \overrightarrow{h}, \overleftarrow{h}, R, u_1$		$u_1, \overleftarrow{h}, \overrightarrow{h}, L, u_1$
$u_1, \lambda, \delta, R, u_1$		$u_1, \lambda, \delta, R, u_1$
		$u_1, \delta, \lambda, L, u_1$

In Cycle 1 the leftmost block of TRs (above) reads the encoded current state. The rightmost block scans left and neutralises markers to index the next ETR. The middle block decides when the cycle is complete. In state u_1 $U_{3,11}$ scans the encoded current state from left to right; each $\overleftarrow{b}$ is changed to $\overleftarrow{e}$ and $U_{3,11}$ then enters state u_2 to see if it is finished reading the encoded current state and encoded read symbol. $U_{3,11}$ is simulating TR $t_{1,1}$ which is encoded by $\mathcal{E}(t_{1,1})$. Hence we have replaced the shorthand notation $\mathcal{E}$ with the word $e^{15}h^2$ defined by $\mathcal{E}(t_{1,1})$. The word $e^{15}h^2$ appears in the location defined by Equation (8). After the initial configuration we have:

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{\overleftarrow{e}\underline{\overleftarrow{a}^{14}}\overleftarrow{b}\,\overleftarrow{b}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{\overleftarrow{e}\,\overleftarrow{e}^{14}\underline{\overleftarrow{b}}\,\overleftarrow{b}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\underline{\overleftarrow{b}}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

The leftmost $\overleftarrow{b}$ is changed to an $\overleftarrow{e}$. $U_{3,11}$ then moves right to test if it is finished reading the encoded current state. If not, $U_{3,11}$ reads another $\overleftarrow{b}$, then scans left in state u_1 and neutralises the rightmost λ marker:

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\underline{\lambda}\,\overrightarrow{e}\,\overline{\overrightarrow{e}^{15}\overrightarrow{e}\,\overleftarrow{b}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\delta\underline{\overrightarrow{e}}\,\overline{\overrightarrow{e}^{15}\overrightarrow{e}\,\overleftarrow{b}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

Having neutralised a λ, $U_{3,11}$ scans right in state u_1 searching for the next $\overleftarrow{b}$.

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\underline{\overleftarrow{b}}}\,\boldsymbol{\overleftarrow{b}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

The neutralisation process is repeated until the end of this cycle. Thus the number of $\overleftarrow{b}$ symbols index the next ETR to be executed. $U_{3,11}$ will know it has finished reading the encoded current state and read symbol when $U_{3,11}$ reads a $\overleftarrow{b}$ in state u_1, rights shifts to test for the end of the encoded current state, and reads an $\overleftarrow{a}$ in state u_2.

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\overleftarrow{e}}\,\underline{\boldsymbol{\overleftarrow{b}}}\,\boldsymbol{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\overleftarrow{e}}\,\boldsymbol{\overleftarrow{e}}\,\underline{\boldsymbol{\overleftarrow{a}}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\overleftarrow{e}\,\underline{\overleftarrow{e}}}\,\gamma\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(I)}$$

In configuration (I) above $U_{3,11}$ has entered Cycle 2. Also, the overlined region is now extended to include the encoded read symbol as this has been read and thus recorded in the same manner as the encoded current state.

Cycle 2 (Print ETR)

$u_2, \overleftarrow{a}, \gamma, L, u_2$	$u_1, \overrightarrow{e}, \overleftarrow{e}, R, u_1$	$u_3, \overrightarrow{e}, \overleftarrow{e}, R, u_3$
$u_2, e, \overleftarrow{e}, R, u_1$	$u_1, \overrightarrow{h}, \overleftarrow{h}, R, u_1$	$u_3, \overrightarrow{h}, \overleftarrow{h}, R, u_3$
$u_2, h, \overleftarrow{h}, R, u_3$	$u_1, \lambda, \delta, R, u_1$	$u_3, \overleftarrow{e}, \gamma, L, u_2$
$u_2, \overleftarrow{e}, \overrightarrow{e}, L, u_2$	$u_1, \gamma, \overleftarrow{a}, L, u_3$	$u_3, \lambda, \delta, R, u_3$
$u_2, \overleftarrow{h}, \overrightarrow{h}, L, u_2$		$u_3, \gamma, \overleftarrow{b}, L, u_3$
$u_2, \lambda, \lambda, R, u_2$		
$u_2, \delta, \lambda, L, u_2$		

This cycle copies an ETR to $\widehat{M}$'s tape head position. The leftmost block scans left and records the next symbol of the ETR to be printed. The two right blocks scan right and print the appropriate symbol. In the configurations below, $U_{3,11}$ scans left until a h is read. Then $U_{3,11}$ right shifts and records this h by entering u_3.

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}hh\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{17}\underline{\overleftarrow{e}}}\gamma\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h\underline{h}\lambda\overrightarrow{\mathcal{E}'}\lambda\overrightarrow{e}\,\overline{\overrightarrow{e}^{17}\overrightarrow{e}}\gamma\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h\overleftarrow{h}\,\underline{\lambda}\overrightarrow{\mathcal{E}'}\lambda\overrightarrow{e}\,\overline{\overrightarrow{e}^{17}\overrightarrow{e}}\gamma\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$U_{3,11}$ now scans right until it reads a γ and prints the recorded symbol.

$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h\overleftarrow{h}\,\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{17}\overleftarrow{e}}\underline{\gamma}\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h\overleftarrow{h}\,\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\overleftarrow{e}^{17}\underline{\overleftarrow{e}}\,\overleftarrow{b}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h\overleftarrow{h}\,\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\,\overline{\underline{\overleftarrow{e}^{17}}\gamma\overleftarrow{b}}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

This printing process is iterated until $U_{3,11}$ is finished printing the ETR. The completion of this process occurs on reading a λ in state u_2:

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\underline{\lambda}\overrightarrow{e}^{15}\overrightarrow{h}^2\lambda\overrightarrow{\mathcal{E}'}\lambda\overrightarrow{e}\,\overline{\overrightarrow{e}\gamma\overleftarrow{a}^{15}\overleftarrow{b}^2}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda\underline{\overrightarrow{e}}^{15}\overrightarrow{h}^2\lambda\overrightarrow{\mathcal{E}'}\lambda\overrightarrow{e}\,\overline{\overrightarrow{e}\gamma\overleftarrow{a}^{15}\overleftarrow{b}^2}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 3 (Restore tape)

$$
\begin{array}{l}
u_2, \overrightarrow{e}, e, R, u_2 \\
u_2, \overrightarrow{h}, h, R, u_2 \\
u_2, \lambda, \lambda, R, u_2 \\
u_2, \gamma, \overleftarrow{h}, R, u_3
\end{array}
$$

These TRs restore M's simulated tape and encoded table of behaviour. This cycle is entered from Cycle 2 (Print ETR). In Cycle 3, $U_{3,11}$ moves right restoring each $\overrightarrow{e}$ to an e and each $\overrightarrow{h}$ to a h. This continues until $U_{3,11}$ reads a γ, switching $U_{3,11}$'s control to u_3. Thus the configuration:

$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{e\underline{\gamma}\overleftarrow{a}\,\overleftarrow{a}^{14}\overleftarrow{b}^2}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

becomes:

$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{e\overleftarrow{h}\,\underline{\overleftarrow{a}}\,\overleftarrow{a}^{14}\overleftarrow{b}^2}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 4 (Choose read or write symbol)

$$
\begin{array}{l}
u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3 \\
u_3, \overleftarrow{b}, e, R, u_1 \\
u_3, e, e, R, u_1 \\
u_3, h, \overleftarrow{a}, L, u_1 \\
u_3, \overleftarrow{h}, \overleftarrow{a}, L, u_3 \\
u_3, \lambda, \delta, R, u_3 \\
u_3, \delta,
\end{array}
$$

This cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if $U_{3,11}$ is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if $U_{3,11}$ is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state. Case (ii) follows:

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda ee\overline{\underline{\overleftarrow{h}}\,\overleftarrow{a}\,\overleftarrow{a}^{14}\overleftarrow{b}^{2}}\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\overline{\underline{e}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{a}^{14}\overleftarrow{b}^{2}}\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda ee\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}^{14}\overleftarrow{b}^{2}}\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^3\lambda e^{15}h^2\lambda\mathcal{E}'\lambda e\boldsymbol{e}\boldsymbol{\overleftarrow{e}}\,\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}^{14}\overleftarrow{b}^{2}}\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(II)}$$

In configuration (II) above we have shortened the overlined region; the two symbols $e\overleftarrow{e}$ to the left of M_1's encoded current state encode the write symbol 0.

The example simulation of TR $t_{1,1} = (q_1, 1, 0, R, q_1)$ is now complete. As $U_{3,11}$ simulates M_1 the encoded tape contents to the left of the simulated tape head is encoded as e and h symbols (i.e. $\widehat{0} = ee$ and $\widehat{1} = he$). The contents to the right is encoded as $\overleftarrow{a}$ symbols and $\overleftarrow{b}$ symbols (as in Definition 1). This is not a problemma as $U_{3,11}$ simulates halting by moving the simulated tape head to the left end of the tape. As a result the entire encoded tape contents of the TM are to the right of the tape head and so are represented by $\overleftarrow{a}$ and $\overleftarrow{b}$ symbols.

In configuration (II) above the encoded write symbol $\widehat{0}$ has been written as the string $e\overleftarrow{e}$ and will become ee after the next ETR has executed. The new encoded current state satisfies Definition 4. The new encoded current state (M_1's simulated tape head) is configured so $U_{3,11}$ reads the next encoded read symbol to the right when searching for the next ETR. The $\overleftarrow{a}$ that signals the end of the encoded current state is provided by the next encoded read symbol $\widehat{0}$.

Remark 1. If the first read symbol of Example 2 is changed from a $\widehat{1}$ to a $\widehat{0}$, then one less $\overleftarrow{b}$ is read when indexing the next ETR. This indexes the rightmost (rather than the second from the right) ETR.

4 Proof of Correctness of $U_{3,11}$

In this section we prove that $U_{3,11}$ correctly simulates a number of the possible types of TRs. We then extend these cases to all cases thus proving the correctness of $U_{3,11}$'s computation.

Lemma 1. *Given a valid initial configuration of $U_{3,11}$, the encoded start state indexes the ETR defined by $\mathcal{E}(t_{1,1})$ if M's read symbol is 1 and $\mathcal{E}'(f,t_{1,0})$ if M's read symbol is 0.*

Proof. The encoded start state contains exactly 2 $\overleftarrow{b}$ symbols. ¿From Example 2 when $U_{3,11}$ reads a $\widehat{1}$ in state $\widehat{q_1}$ it neutralises two λ markers thus locating the second ETR from the right. By Definition 2 and Equation (2) this ETR is defined by $\mathcal{E}(t_{1,1})$. From Remark 1 and Example 2 when $U_{3,11}$ reads a $\widehat{0}$ in state $\widehat{q_1}$ it neutralises one λ, thus indexing the rightmost ETR defined by $\mathcal{E}'(f,t_{1,0})$. □

Example 3 ($U_{3,11}$'s simulation of TR $t_{1,0}=(q_1,0,1,R,q_2)$ in M_1). In this example $U_{3,11}$ is reading a $\widehat{0}$ *after* a right shift. The right shift was given by the simulation of $t_{1,1}=(q_1,1,0,R,q_1)$ in Example 2. This unique case involves two steps, executing an ETR′ and then an ETR. The execution of an ETR′ is represented by parts (a) and (b) of Fig. 3 and the execution of the subsequent ETR is represented by parts (c) and (d) of Fig. 3.

We take the last configuration of Example 2, with the encoded read symbol $\widehat{0}=\overleftarrow{a}\,\overleftarrow{a}$ to the right of the encoded current state. Substituting the appropriate ETR′ $e^{12}h^4$ from Equation (8) gives:

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda ee\overleftarrow{e}\,\overline{\underline{\overleftarrow{a}^{15}}\,\overleftarrow{b}\,\overleftarrow{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\delta\overleftarrow{e}\,\overleftarrow{e}\,\overleftarrow{e}\,\overline{\overleftarrow{e}^{15}\overleftarrow{e}\,\underline{\overleftarrow{e}}\gamma}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

In the configuration immediately above we have reached the end of Cycle 1 (Index next ETR). One λ has been changed to a δ indexing the ETR′ $e^{12}h^4$. The $\overleftarrow{b}\,\overleftarrow{a}$ that signaled the end of Cycle 1 was provided by the rightmost $\overleftarrow{b}$ of the encoded current state and the leftmost $\overleftarrow{a}$ of the encoded read symbol. Thus, only the leftmost $\overleftarrow{a}$ of $\widehat{0}=\overleftarrow{\boldsymbol{a}}\,\overleftarrow{\boldsymbol{a}}$ was read and this is sufficient to distinguish $\widehat{0}$ from $\widehat{1}=\overleftarrow{b}\,\overleftarrow{a}$. However the overlined region does not cover the entire encoded read symbol which is why an ETR′ executes before an ETR in this unique case. Skipping to the end of Cycle 2 (Print ETR) gives:

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\underline{\lambda}\,\overrightarrow{e}^{12}\overrightarrow{h}^{4}\lambda\overrightarrow{e}\,\overrightarrow{e}\,\overrightarrow{e}\,\overline{\overrightarrow{e}\gamma\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda eeee\overline{\underline{\gamma}\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda eeee\overline{\overleftarrow{h}\,\underline{\overleftarrow{a}}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda eeee\overline{\underline{\overleftarrow{h}}\,\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda eee\overline{\underline{e}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^4\lambda e^{12}h^4\lambda eeee\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

At this point $U_{3,11}$ has executed the ETR′. $U_{3,11}$ now executes the ETR that represents the second step of the simulation of TR $t_{1,0}$. This ETR is defined by $\mathcal{E}(t_{1,0})$. Substituting the ETR $he^{10}h^7$ from Equation (8) into the configuration immediately above gives:

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^7\lambda\mathcal{E}\lambda\mathcal{E}'\lambda eeee\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}\,\overleftarrow{a}^{11}\overleftarrow{b}^{4}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

We now skip to the end of Cycle 1 (Index next ETR) giving:

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^7\delta\overleftarrow{\mathcal{E}}\,\delta\overleftarrow{\mathcal{E}'}\,\delta\overleftarrow{e}\,\overleftarrow{e}\,\overleftarrow{e}\,\overline{\overleftarrow{e}^{18}\underline{\overleftarrow{\boldsymbol{a}}}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^7\delta\overleftarrow{\mathcal{E}}\,\delta\overleftarrow{\mathcal{E}'}\,\delta\overleftarrow{e}\,\overleftarrow{e}\,\overleftarrow{e}\,\overline{\underline{\overleftarrow{e}}^{18}}\gamma\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(III)}$$

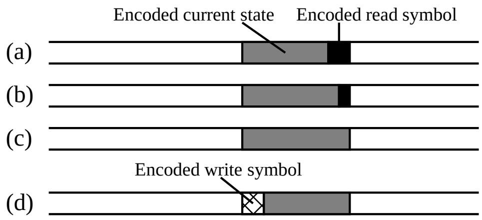

Fig. 3. Right shift simulation (special case). The Encoded current state marks the location of M's simulated tape head. The configurations given in (a), (b) and (c) represent the reading of the encoded current state and an encoded 0 following a right shift. (a) Encoded configuration before beginning the TR simulation. (b) Intermediate configuration after the encoded current state and first symbol of the encoded read symbol have been read. (c) Intermediate configuration immediately after the remainder of the encoded read symbol has been read. (d) Configuration immediately after the simulated right shift.

Notice that the ETR is indexed by neutralising 3 λ markers and the second part of the $\widehat{0}$ has been consumed in this process. We now skip to the end of Cycle 3 (Restore tape) and illustrate a $\widehat{1}$ being written to the left of the encoded current state:

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\underline{\lambda}\overrightarrow{h}\overrightarrow{e}^{10}\overrightarrow{h}^{7}\lambda\overrightarrow{\mathcal{E}}\lambda\overrightarrow{\mathcal{E}'}\lambda\overrightarrow{e}\overrightarrow{e}\overrightarrow{e}\overline{\gamma\overleftarrow{b}\overleftarrow{a}^{10}\overleftarrow{b}^{7}}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^{7}\lambda\mathcal{E}\lambda\mathcal{E}'\lambda eee\overline{\underline{\gamma}\overleftarrow{b}\overleftarrow{a}^{10}\overleftarrow{b}^{7}}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^{7}\lambda\mathcal{E}\lambda\mathcal{E}'\lambda eee\overline{\overleftarrow{h}\underline{\overleftarrow{b}}\overleftarrow{a}^{10}\overleftarrow{b}^{7}}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^2(\lambda\mathcal{E})^2\lambda he^{10}h^{7}\lambda\mathcal{E}\lambda\mathcal{E}'\lambda eee\overleftarrow{\boldsymbol{h}}\boldsymbol{e}\overline{\underline{\overleftarrow{a}}^{10}\overleftarrow{b}^{7}}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^{\omega}$$

In the configuration immediately above the write symbol is positioned to the left of the new encoded current state. Recall that to the left of the simulated tape head the symbol 1 is encoded as he. The $\overleftarrow{h}$ will become a h after execution of the next ETR. The new encoded current state satisfies Definition 4 and the simulation of TR $t_{1,0} = (q_1, 0, 1, R, q_2)$ is complete.

Lemma 2. *Given a valid configuration of $U_{3,11}$, the encoded current state $\widehat{q_x}$ and encoded read symbol $\widehat{\sigma_1}$ index the ETR $\mathcal{E}(t_{x,\sigma_1})$.*

Proof. $\widehat{M}$ is a list of ETRs, five ETRs for each state (pair of TRs) in M. The number of $\overleftarrow{b}$ symbols, in the encoded current state $\widehat{q_x}$, index the next ETR to be executed. In the encoding, the function $b(\cdot)$ determines the number of $\overleftarrow{b}$ symbols in the next encoded current state. The function $b(\cdot)$ is defined as a summation over $g(\cdot)$ for j, $1 \leq j \leq x$.

For each $j < x$, the function $g(\cdot)$ always has value 5, hence there are at least $\boldsymbol{(x-1)5}$ juxtaposed $\overleftarrow{b}$ symbols in $\widehat{q_x}$. The state q_x is encoded using five ETRs. When $j = x$, then $g = 0$ or $g = 3$; giving a total number of $\overleftarrow{b}$ symbols that point to the first or fourth of these five ETRs.

Any encoded current state $\widehat{q_x}$, was established by execution of an ETR r. The ETR r encodes shift direction D_r and next state q_x. The location of the ETR that is indexed by $\widehat{q_x}$ is dependant on the shift direction D_r of r. When $D_r = L$ and $j = x$ then $g(\cdot) = 3$;

when this 3 is added to $\boldsymbol{(x-1)5}$ this indexes the 4^{th} ETR (from right) of the ETRs for q_x. Using this value $((x-1)5+3)$ we get Case A and Case B below. For clarity at this point note that D_r is the shift direction of the ETR r that *established* $\widehat{q_x}$ and $\widehat{\sigma_1}$ is the read symbol that is read with $\widehat{q_x}$ to index the *next* ETR $\mathcal{E}(t_{x,\sigma_1})$.

Case A: $(D_r = L, \sigma_1 = 0)$. $\widehat{0} = \overleftarrow{a}\overleftarrow{a}$ adds no extra $\overleftarrow{b}$ symbols to the list of $\overleftarrow{b}$ symbols in $\widehat{q_x}$, thus the number of $\overleftarrow{b}$ symbols is given by $g(\cdot)$ alone and indexes the 4^{th} ETR (from right). By Equation (2) this is $\mathcal{E}(t_{x,0})$.

Case B: $(D_r = L, \sigma_1 = 1)$. $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$ adds one extra $\overleftarrow{b}$ to the list of $\overleftarrow{b}$ symbols in $\widehat{q_x}$, thus indexing the 5^{th} ETR (from right). By Equation (2) this is $\mathcal{E}(t_{x,1})$.

When $D_r = R$ and $j = x$ then $g(\cdot) = 0$. Adding this 0 to $\boldsymbol{(x-1)5}$ we get Case C and Case D.

Case C: $(D_r = R, \sigma_1 = 1)$. $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$ adds one extra $\overleftarrow{b}$ to the list of $\overleftarrow{b}$ symbols in $\widehat{q_x}$, thus indexing the 2^{th} ETR (from right). By Equation (2) this is $\mathcal{E}(t_{x,1})$.

Case D: $(D_r = R, \sigma_1 = 0)$. Case D is a unique case in which $U_{3,11}$ simulates a TR t with read symbol 0, immediately after a right shifting TR $t^{R,x}$ (i.e. $t^{R,x} \vdash t$). In such a case t is encoded as 2 ETRs using $\mathcal{E}$ and $\mathcal{E}'$. The encoded read symbol $\widehat{0} = \overleftarrow{a}\overleftarrow{a}$ adds no extra $\overleftarrow{b}$ symbols thus indexing the rightmost ETR, denoted ETR$'$. This ETR$'$ is given by the function $\mathcal{E}'$ and establishes an *intermediate* encoded current state $\widehat{q_x}'$ that indexes another ETR that in turn completes the simulation of t. This other ETR is positioned 2 ETRs to the left of the ETR$'$. Hence in Equation (4), $t^{R,x}$ is passed to $b(\cdot)$ as a parameter (instead of t) and $\mathcal{E}'$ adds 2 $\overleftarrow{b}$ symbols to index the ETR 2 places to the left of ETR$'$. By Equation (2) this is $\mathcal{E}(t_{x,0})$. $\square$

Examples 2 and 3 give simulations of right shifting TRs with the later case covering the special case of reading a 0 after a right shift. Example 4 gives the simulation of a left shifting TR.

Example 4 ($U_{3,11}$'s simulation of TR $t_{2,1} = (q_2, 1, 1, L, q_3)$ in M_1). We take the last configuration of Example 3, with $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$ to the right of the encoded current state. Substituting the appropriate ETR from Equation (8) gives:

$$\boldsymbol{u_1}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda eee\overleftarrow{h}\,e\,\overline{\underline{\overleftarrow{a}}^{10}\overleftarrow{b}^{7}}\boldsymbol{\overleftarrow{b}}\boldsymbol{\overleftarrow{a}}\overleftarrow{a}^{\omega}$$

We now skip to the end of Cycle 1 (Index next ETR) giving:

$$\boldsymbol{u_2}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\delta\overleftarrow{\mathcal{E}'}(\delta\overleftarrow{\mathcal{E}})^4\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{e}\overleftarrow{e}\overleftarrow{e}\overleftarrow{h}\overleftarrow{e}\overline{\underline{\overleftarrow{e}}^{18}}\gamma\overleftarrow{a}^{\omega} \qquad \text{(IV)}$$

Notice that the ETR is indexed by neutralising 7 λ markers reading the $\widehat{1}$ in this process. Next the ETR $eh^{15}ehe$ is printed and we skip to the end of Cycle 2 (Print ETR):

$$\boldsymbol{u_2}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\underline{\lambda}\overrightarrow{e}\overrightarrow{h}^{15}\overrightarrow{e}\overrightarrow{h}\overrightarrow{e}\delta\overrightarrow{\mathcal{E}'}(\delta\overrightarrow{\mathcal{E}})^4\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{e}\overrightarrow{e}\overrightarrow{e}\overrightarrow{h}\gamma\overline{\overleftarrow{a}\overleftarrow{b}^{15}\overleftarrow{a}\overleftarrow{b}}\overleftarrow{a}\overleftarrow{a}^{\omega}$$

Skipping to the end of Cycle 3 (Restore tape) gives:

$$\boldsymbol{u_2}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda eeeh\underline{\gamma}\overline{\overleftarrow{a}\overleftarrow{b}^{15}\overleftarrow{a}}\boldsymbol{\overleftarrow{b}}\boldsymbol{\overleftarrow{a}}\overleftarrow{a}^{\omega} \qquad \text{(V)}$$

In configuration (V) above the correct write symbol ($\widehat{1} = \overleftarrow{b}\overleftarrow{a}$) has been placed to the right of the encoded current state. The new encoded current state satisfies Definition 4 and the simulation of TR $t_{1,1} = (q_2, 1, 1, L, q_3)$ is complete.

Remark 2. We show how $U_{3,11}$ reads an encoded read symbol *following a left shift*. In this case the encoded read symbol is to the left of the encoded current state. Immediately after configuration (V) of Example 4 we would get:

$$\boldsymbol{u_3}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda eee\boldsymbol{h}\overleftarrow{\boldsymbol{h}}\overline{\underline{\overleftarrow{a}}\,\overleftarrow{b}^{15}\overleftarrow{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda eee\boldsymbol{h}\underline{\overleftarrow{\boldsymbol{h}}}\overline{\overleftarrow{a}\,\overleftarrow{b}^{15}\overleftarrow{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda eee\underline{\boldsymbol{h}}\overleftarrow{\boldsymbol{a}}\overline{\overleftarrow{a}\,\overleftarrow{b}^{15}\overleftarrow{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(VI)}$$

$$\boldsymbol{u_1}(\lambda\mathcal{E})^4\lambda\mathcal{E}'(\lambda\mathcal{E})^3\lambda eh^{15}ehe\lambda\mathcal{E}'(\lambda\mathcal{E})^4\lambda\mathcal{E}'\lambda ee\underline{e}\overline{\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}^{15}\overleftarrow{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(VII)}$$

In configuration (VII) above the overlined region is extended as the encoded read symbol has been read. $U_{3,11}$ has begun to index the next ETR and is moving to the left to neutralise a λ. The rightmost symbol of the encoded read symbol (which for left shifts is always e) was previously overwritten with a γ and this was eventually changed to a $\overleftarrow{h}$. Only the leftmost symbol of the encoded read symbol must be recorded. If the read symbol was a $\widehat{0} = ee$ then $U_{3,11}$'s tape head would have read an e instead of a h in configuration (VI) above, sending $U_{3,11}$'s tape head right instead of left. This would result in one less λ being neutralised. This process records the difference in the encoded read symbols ee and he.

Continuing from configuration (VII) immediately after the next ETR has been indexed, we have the following configuration:

$$\boldsymbol{u_2}\lambda eh^{15}ehe(\delta\overleftarrow{\mathcal{E}})^3\delta\overleftarrow{\mathcal{E}'}((\delta\overleftarrow{\mathcal{E}})^4\delta\overleftarrow{\mathcal{E}'})^2\delta\overleftarrow{e}\,\overleftarrow{e}\,\overleftarrow{e}\,\overline{\underline{\overleftarrow{e}}^{18}}\gamma\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega} \qquad \text{(VIII)}$$

Lemma 3. *Given a valid configuration with the encoded current state $\widehat{q_{|Q|}}$ then $U_{3,11}$ halts.*

Proof. Recall from Section 2.3 that for all M the TRs for the halt state $q_{|Q|}$ are left shifting and have $q_{|Q|}$ as the next state. Thus when $\widehat{M}$ enters $\widehat{q_{|Q|}}$ $U_{3,11}$ then simulates repeated left shifts. These left shifts continue until the left end of M's simulated tape is reached. When the simulated tape head is attempting to left shift at the left end of the simulated tape then $U_{3,11}$'s always has the following configuration:

$$\boldsymbol{u_2}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\lambda\underline{\gamma}\overline{\overleftarrow{a}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\lambda\overleftarrow{h}\overline{\underline{\overleftarrow{a}}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\lambda\underline{\overleftarrow{h}}\overline{\overleftarrow{a}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\underline{\lambda}\overleftarrow{a}\overline{\overleftarrow{a}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\delta\underline{\overleftarrow{a}}\overline{\overleftarrow{a}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

$$\boldsymbol{u_3}(\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}\lambda\mathcal{E}')^3\underline{\delta}\overleftarrow{a}\overline{\overleftarrow{a}\,\overleftarrow{a}^*\overleftarrow{b}^2\overleftarrow{b}^*\overleftarrow{a}}(\overleftarrow{b}\,\overleftarrow{a}\cup\overleftarrow{a}\,\overleftarrow{a})^*\overleftarrow{a}^{\omega}$$

There is no TR for 'when in state u_3 read a δ' in $U_{3,11}$ so the simulation halts. $\square$

Lemma 4. *Given a valid initial configuration of $U_{3,11}$, immediately after the first ETR of a computation is indexed, the overlined region is of the form $\overleftarrow{e}^{5|Q|+3}\gamma$.*

Proof. The encoded start state $\widehat{q_1}$ in an initial configuration is of the form $\overleftarrow{a}^{5|Q|}\overleftarrow{b}^2$. Example 2 gives the case of reading a $\widehat{1}$ in the encoded start state $\widehat{q_1}$. In this case $\widehat{q_1}$ and $\widehat{1}$ have both been read, that is the $\overleftarrow{b}$ and $\overleftarrow{a}$ symbols of $\widehat{q_1}$ and $\widehat{1}$ have been changed to $\overleftarrow{e}$ symbols and the rightmost $\overleftarrow{a}$ is replace by γ. This gives and overlined region of $\overleftarrow{e}^{5|Q|+3}\gamma$.

The other case is reading a $\widehat{0}$ in state $\widehat{q_1}$. By definition the $\widehat{0}$ is located immediately to the right of $\widehat{q_1}$. When reading a $\widehat{0}$ on the right there are two steps; executing an ETR′ and then an ETR. From Lemma 1 we know that $\mathcal{E}'(f, t_{1,0})$ is indexed. From Example 3 we know that the entire $\widehat{0}$ is read after a subsequent ETR is indexed immediately following execution of an $\mathcal{E}'$. Thus the $\overleftarrow{b}$ and $\overleftarrow{a}$ symbols of $\widehat{q_1}$ and $\widehat{0}$ have been changed to $\overleftarrow{e}$ symbols and the rightmost $\overleftarrow{a}$ is replace by γ. This gives and overlined region of $\overleftarrow{e}^{5|Q|+3}\gamma$. □

Lemma 5. *$U_{3,11}$ simulates any TR of any deterministic TM M.*

Proof. The proof is by induction on the form of the overlined region. The base case is given by Lemma 4; after the first ETR is indexed then the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$.

We will show that immediately after any ETR is indexed, the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$.

Assume that the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ immediately after indexing an ETR ξ_1 in the simulation of timestep i of M's computation. Let ξ_2 be the ETR that is executed immediately after ξ_1. We now show that the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ immediately after indexing ξ_2 in the simulation of timestep $i+1$.

The four cases of ETRs are defined by Equation (3). In Examples 2, 3 and 4, three of these cases are shown to execute correctly on an overlined region of the form $\overleftarrow{e}^{5|Q|+3}\gamma$. We use Example 4 to verify the remaining case (left shift write a 0) by substitution of the ETR defined by case 4 of Equation (3) with the ETR defined by case 3.

We show that the four cases given above for ξ_1 executing on an overlined region of $\overleftarrow{e}^{5|Q|+3}\gamma$, will result in the overlined region being $\overleftarrow{e}^{5|Q|+3}\gamma$ immediately after ξ_2 is indexed. When ξ_2 is indexed simulation of timestep i will have been completed and simulation of timestep $i+1$ will have begun.

Case 1 of Equation (3)**:** Examples 2 and 3 verify Case 1. In configuration (I) above (in simulation of timestep i) the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ and the ETR ξ_1 indexed is defined by Case 1 of Equation (3). In configuration (III) above (in simulation of timestep $i+1$) the next ETR ξ_2 has been indexed and the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$.

Case 2 of Equation (3)**:** Examples 3 and 4 verify Case 2. In configuration (III) above (in simulation of timestep i) the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ and the ETR ξ_1 indexed is defined by Case 2 of Equation (3). In configuration (IV) above (in simulation of timestep $i+1$) the next ETR ξ_2 has been indexed and the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$.

Cases 4 of Equation (3)**:** Example 4 and configuration (VIII) verify Case 4. In configuration (IV) above (in simulation of timestep i) the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ and the ETR ξ_1 indexed is defined by Case 4 of Equation (3). In configuration (VIII) above (in simulation of timestep $i+1$) the next ETR ξ_2 has been indexed and the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$.

Cases 3 of Equation (3)**:** Case 4 also verifies Case 3 by substitution of the ETR defined by Case 4 of Equation (3) with the ETR defined by Case 3.

We have shown that the overlined region is $\overleftarrow{e}^{5|Q|+3}\gamma$ immediately after any ETR is indexed. ¿From Examples 2, 3 and 4, each ETR executes correctly on an overlined region of $\overleftarrow{e}^{5|Q|+3}\gamma$. This establishes the correct simulated tape head location, encoded write symbol and an encoded current state that satisfies Definition 4. By Lemmas 1 and 2 the encoded current state indexes the correct ETR. Due to the relative lengths of the encoded current state and overlined region the above mention examples generalise to any TR of any TM M. $\square$

Let M be a deterministic TM with $|Q|$ states and time complexity $T(n)$ on input length n.

Theorem 1. *$U_{3,11}$ simulates any TM M in space $O(|Q|^2 + T(n))$ and time $O(|Q|^3 T(n) + |Q| T^2(n))$.*

Proof. By the previous lemma $U_{3,11}$ simulates any TR. Thus given a valid encoding of M's initial configuration (Definition 2), $U_{3,11}$ simulates the sequence of TRs in M's computation. From Lemma 3 when $U_{3,11}$ simulates the halting state of M, $U_{3,11}$'s tape head returns to the left end of M's encoded output and halts. The encoded output is easily decoded via Definition 1.

(Space). At time $T(n)$ the space used by M is bounded by $T(n)$. Simulator $U_{3,11}$ requires space $O(|Q|^2 + T(n))$, where $O(|Q|^2)$ space is required to store M as the word $\widehat{M}$ and $O(T(n))$ space is required to store M's encoded tape after $T(n)$ simulated steps.

(Time). Simulating a TR involves 4 cycles. (1) Index an ETR by neutralising $O(|Q|)$ of the λ markers: $O(|Q|^3 + |Q|T(n))$ steps. (2) Copy an ETR of length $O(|Q|)$ from $\widehat{M}$ to the encoded current state location: $O(|Q|^3 + |Q|T(n))$ steps. (3) Restore $U_{3,11}$'s tape contents: $O(|Q|^2 + T(n))$ steps. (4) Complete execution of ETR: a small constant number of steps. Thus $U_{3,11}$ requires $O(|Q|^3 + |Q|T(n))$ time to simulate a single step of M, and worst case $O(|Q|^3 T(n) + |Q| T^2(n))$ time to simulate the entire computation of M. $\square$

This result holds for more general definitions of TMs. For example, let M' be a deterministic multitape TM with bi-infinite tapes and greater than two symbols. M' would be converted to a two symbol, one-way-infinite single tape TM M. The number of states in M would be only a constant times greater than the state-symbol product of M', also M would be at worst polynomially slower than M'. Thus, $U_{3,11}$ simulates M' in polynomial time. We get the following immediate corollary.

Corollary 1. *There are polynomial time UTMs in UTM(m, n) for all $m \geq 3$, $n \geq 11$.*

5 Polynomial time Curve

In the section we further extend our result from the previous section by finding small polynomial time UTMs in other classes. Thus we establish a polynomial time curve of small UTMs similar to what Rogozhin [5] has done with Minsky's [4] exponential UTM in UTM(7,4).

All UTMs in this paper use the same basic algorithm as $U_{3,11}$. The proof of correctness given for $U_{3,11}$ can be applied to the remaining machines in a straightforward way, so we do not restate them. The encoding of the input and operation of these UTMs is the same as $U_{3,11}$ unless noted otherwise. Each UTM makes use of specially tailored $\mathcal{E}$ and $\mathcal{E}'$ functions.

5.1 Construction of $U_{6,6}$

For $U_{6,6}$ the start state of $\widehat{M}$ is encoded as $\widehat{q_1} = \overleftarrow{a}^{5|Q|}\overleftarrow{b}^{2}$. The encoding of the current state is of the form $\overleftarrow{a}^{*}\overleftarrow{b}^{2}\overleftarrow{b}^{*}\{\overleftarrow{a} \cup \epsilon\}$ and is of length $5|Q|+2$.

Let $t = (q_x, \sigma_1, \sigma_2, D, q_y)$ be a fixed TR in M, then t is encoded via $\mathcal{P}$ using the function $\mathcal{E}$ on its own or in conjunction with $\mathcal{E}'$ where

$$\mathcal{E}(t)=\begin{cases} \overleftarrow{b}^{b(t)}\overleftarrow{a}^{a(,t)-3}\overleftarrow{b} & \text{if } D=R, \sigma_2=0,\\ \overleftarrow{b}^{b(t)}\overleftarrow{a}^{a(,t)+1} & \text{if } D=R, \sigma_2=1,\\ \overleftarrow{a}\overleftarrow{a}\overleftarrow{a}\overleftarrow{b}^{b(t)}\overleftarrow{a}^{a(,t)-2}\overleftarrow{b} & \text{if } D=L, \sigma_2=0,\\ \overleftarrow{a}\overleftarrow{b}\overleftarrow{a}\overleftarrow{b}^{b(t)}\overleftarrow{a}^{a(,t)-2}\overleftarrow{b} & \text{if } D=L, \sigma_2=1, \end{cases} \tag{9}$$

and

$$\mathcal{E}'(f,t) = \begin{cases} \overleftarrow{b}^{b(t^{R,x})+2}\overleftarrow{a}^{a(,t^{R,x})-5}\overleftarrow{b} & \text{if } \exists t^{R,x}, q_x \neq q_1,\\ \overleftarrow{b}^{4}\overleftarrow{a}^{5|Q|-5}\overleftarrow{b} & \text{if } q_x = q_1,\\ \epsilon & \text{if } \neg\exists t^{R,x}, q_x \neq q_1, \end{cases} \tag{10}$$

where as before $t^{R,x}$ is any right shifting TR such that $t^{R,x} \vdash t$.

The value of E from Definition 2 for $U_{6,6}$ is $E = \overleftarrow{a}$.

Example 5 (Encoding of TM M_2). Let TM $M_2 = (\{q_1, q_2\}, \{0, 1\}, 0, f, q_1, \{q_2\})$ where f is defined by $(q_1, 0, 0, R, q_1)$, $(q_1, 1, 1, R, q_2)$, $(q_2, 0, 0, L, q_2)$ and $(q_2, 1, 1, L, q_2)$. M_2 is encoded as: $\widehat{M_2} = \lambda\mathcal{P}(f, q_2)\lambda\mathcal{P}(f, q_1)\lambda E$. Substituting the appropriate values from Equation (2) gives

$$\begin{aligned}\widehat{M_2} =&\lambda\mathcal{E}(t_{2,1})\lambda\mathcal{E}(t_{2,0})\lambda\mathcal{E}(t_{2,0})\lambda\mathcal{E}(t_{2,1})\lambda\mathcal{E}'(f, t_{2,0})\\ &\lambda\mathcal{E}(t_{1,1})\lambda\mathcal{E}(t_{1,0})\lambda\mathcal{E}(t_{1,0})\lambda\mathcal{E}(t_{1,1})\lambda\mathcal{E}'(f, t_{1,0})\lambda E\end{aligned}$$

Rewriting this using Equations (9) and (10) gives

$$\begin{aligned}\widehat{M_2} =&\lambda\overleftarrow{a}\overleftarrow{b}\overleftarrow{a}\overleftarrow{b}^{11}\lambda\overleftarrow{a}\overleftarrow{a}\overleftarrow{a}\overleftarrow{b}^{11}\lambda\overleftarrow{a}\overleftarrow{a}\overleftarrow{a}\overleftarrow{b}^{11}\lambda\overleftarrow{a}\overleftarrow{b}\overleftarrow{a}\overleftarrow{b}^{11}\\ &\lambda\overleftarrow{b}^{10}\lambda\overleftarrow{b}^{7}\overleftarrow{a}^{6}\lambda\overleftarrow{b}^{2}\overleftarrow{a}^{7}\overleftarrow{b}\lambda\overleftarrow{b}^{2}\overleftarrow{a}^{7}\overleftarrow{b}\lambda\overleftarrow{b}^{7}\overleftarrow{a}^{6}\lambda\overleftarrow{b}^{4}\overleftarrow{a}^{5}\overleftarrow{b}\lambda\overleftarrow{a}\end{aligned} \tag{11}$$

Definition 7 ($U_{6,6}$). *The TM $U_{6,6}$ is defined as $U_{6,6} = (\{u_1, u_2, u_3, u_4, u_5, u_6\}, \{\overleftarrow{a}, \overleftarrow{b}, \overrightarrow{a}, \overrightarrow{b}, \lambda, \delta\}, \overleftarrow{a}, f, u_1, \{u_3, u_5, u_6\})$ where f is given by the following transition rules.*

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overrightarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \overrightarrow{b}, L, u_3$
$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overrightarrow{a}, \overleftarrow{a}, R, u_2$	$u_3, \overrightarrow{a}$
$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$	$u_2, \overrightarrow{b}, \overleftarrow{b}, R, u_2$	$u_3, \overrightarrow{b}, \overleftarrow{a}, L, u_5$
$u_1, \lambda, \overleftarrow{b}, L, u_2$	$u_2, \lambda, \overleftarrow{a}, L, u_2$	$u_3, \lambda, \delta, R, u_1$
$u_1, \delta, \delta, R, u_1$	$u_2, \delta, \delta, R, u_2$	$u_3, \delta, \delta, L, u_3$
$u_4, \overleftarrow{a}, \overrightarrow{a}, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, L, u_1$	$u_6, \overleftarrow{a}$
$u_4, \overleftarrow{b}, \overrightarrow{b}, L, u_4$	$u_5, \overleftarrow{b}, \overleftarrow{a}, L, u_3$	$u_6, \overleftarrow{b}$
$u_4, \overrightarrow{a}, \overleftarrow{a}, R, u_5$	$u_5, \overrightarrow{a}, \overrightarrow{a}, R, u_2$	$u_6, \overrightarrow{a}, \overleftarrow{a}, R, u_6$
$u_4, \overrightarrow{b}, \overleftarrow{b}, R, u_5$	$u_5, \overrightarrow{b}, \overrightarrow{b}, R, u_1$	$u_6, \overrightarrow{b}, \overleftarrow{b}, R, u_6$
$u_4, \lambda, \lambda, R, u_5$	u_5, λ	$u_6, \lambda, \overrightarrow{b}, R, u_5$
$u_4, \delta, \delta, L, u_4$	$u_5, \delta, \lambda, R, u_6$	$u_6, \delta, \lambda, R, u_6$

Remark 3. There are some minor differences between the operation of $U_{6,6}$ and $U_{3,11}$. The order of symbols in ETRs of $U_{6,6}$ is reversed when compared with ETRs of $U_{3,11}$, assuming $\overleftarrow{a} = e$ and $\overleftarrow{b} = h$. To see this note the difference between Equations (3) and (9). When printing an ETR, $U_{6,6}$ reverses the order so that encoded current states are of the same form as those in $U_{3,11}$. Also M's encoded tape symbols to the left and right of the encoded current state use the *same* encodings ($\widehat{0} = \overleftarrow{a}\,\overleftarrow{a}$ and $\widehat{1} = \overleftarrow{b}\,\overleftarrow{a}$). This is not the case for $U_{3,11}$.

We give an example of $U_{6,6}$ simulating a TR of M_2 from Example 5. As usual the example is broken down into 4 cycles.

Example 6 ($U_{6,6}$'s simulation of TR $t_{1,1} = (q_1, 1, 1, R, q_2)$ from TM M_2). The start state of $U_{6,6}$ is u_1. $U_{6,6}$'s tape head is over the symbol directly to the right of $\widehat{M_2}$ (as in Equation (11)). The input to M_2 is 11 ($\widehat{1} = \overleftarrow{b}\,\overleftarrow{a}$). Thus the initial configuration is:
$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}'})^2\lambda\overleftarrow{a}\,\overline{\underline{\overleftarrow{a}}^{10}\overleftarrow{b}^{2}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 1 (Choose read or write symbol)

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overrightarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \overrightarrow{b}, L, u_3$
$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$		$u_3, \lambda, \delta, R, u_1$
$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$		$u_3, \delta, \delta, L, u_3$
$u_1, \delta, \delta, R, u_1$		

In Cycle 1 the left block of TRs (above) reads the encoded current state. The right block neutralises λ markers to index the next ETR. The neutralisation is done in the usual way each $\overleftarrow{b}$ in the encoded current state causes a λ to be changed to a δ. The middle block decides when the cycle is complete. In state u_1 $U_{6,6}$ scans the encoded current state from left to right; each $\overleftarrow{b}$ is changed to an $\overleftarrow{a}$ and $U_{6,6}$ then enters state u_3 via u_2.

We have replaced the shorthand notation $\overleftarrow{\mathcal{E}}$ with the word $\overleftarrow{b}^7\overleftarrow{a}^6$ defined by $\mathcal{E}(t_{1,1})$. The word $\overleftarrow{b}^7\overleftarrow{a}^6$ appears in the location defined by Equation (11). After the initial configuration we have:
$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overline{\overleftarrow{a}^{10}\underline{\overleftarrow{b}}\,\overleftarrow{b}}\,\overleftarrow{\boldsymbol{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_2}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overline{\overleftarrow{a}^{10}\overleftarrow{a}\,\underline{\overleftarrow{b}}}\,\overleftarrow{\boldsymbol{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_3}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overline{\overleftarrow{a}^{10}\underline{\overleftarrow{a}}\,\overleftarrow{b}}\,\overleftarrow{\boldsymbol{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_3}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\underline{\lambda}\overrightarrow{a}\,\overline{\overrightarrow{a}^{10}\overrightarrow{a}\,\overleftarrow{b}}\,\overleftarrow{\boldsymbol{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\delta\underline{\overrightarrow{a}}\,\overline{\overrightarrow{a}^{10}\overrightarrow{a}\,\overleftarrow{b}}\,\overleftarrow{\boldsymbol{b}}\,\overleftarrow{\boldsymbol{a}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

The neutralisation process continues until $U_{6,6}$ reads the final $\overleftarrow{b}$, rights shifts to test for the end of the encoded current state in u_2, and then reads an $\overleftarrow{a}$. $U_{6,6}$ then knows it has finished reading the encoded current state and read symbol. Skipping to the end of this cycle gives:
$\boldsymbol{u_4}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{a}\,\overline{\overleftarrow{a}^{10}\overleftarrow{a}\,\overleftarrow{a}\,\underline{\overleftarrow{a}}}\lambda\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$U_{6,6}$ has neutralised two λ markers to index the next ETR.

Cycle 2 (Print ETR)

$u_4, \overleftarrow{a}, \overrightarrow{a}, L, u_4$	$u_5, \overrightarrow{a}, \overrightarrow{a}, R, u_2$	$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$
$u_4, \overleftarrow{b}, \overrightarrow{b}, L, u_4$	$u_5, \overrightarrow{b}, \overrightarrow{b}, R, u_1$	$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$	$u_2, \overrightarrow{a}, \overleftarrow{a}, R, u_2$
$u_4, \overrightarrow{a}, \overleftarrow{a}, R, u_5$	$u_5, \delta, \lambda, R, u_6$	$u_1, \lambda, \overleftarrow{b}, L, u_2$	$u_2, \overrightarrow{b}, \overleftarrow{b}, R, u_2$
$u_4, \overrightarrow{b}, \overleftarrow{b}, R, u_5$		$u_1, \delta, \delta, R, u_1$	$u_2, \lambda, \overleftarrow{a}, L, u_2$
$u_4, \lambda, \lambda, R, u_5$			$u_2, \delta, \delta, R, u_2$
$u_4, \delta, \delta, L, u_4$			

This cycle copies an ETR to M's simulated tape head position. The leftmost block scans left and locates the next symbol of the ETR to be printed. The second block from the left records the symbol to be printed or ends the cycle. The rightmost two blocks scan right and print the appropriate symbol. In the configurations below, $U_{6,6}$ scans left until a λ is read. Then $U_{6,6}$ right shifts and records the symbol read by entering state u_1 or u_2.

$\boldsymbol{u_4}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\underline{\lambda}\,\overrightarrow{b}\,\overrightarrow{b}\,\overrightarrow{b}^5\overrightarrow{a}^6\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\overrightarrow{a}^{12}\overrightarrow{a}\lambda}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\underline{\overrightarrow{b}}\,\overrightarrow{b}\,\overrightarrow{b}^5\overrightarrow{a}^6\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\overrightarrow{a}^{12}\overrightarrow{a}\lambda}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overrightarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}^5\overleftarrow{a}^6\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{a}\,\overline{\overleftarrow{a}^{12}\overleftarrow{a}\underline{\lambda}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_2}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overrightarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}^5\overleftarrow{a}^6\delta\overleftarrow{\mathcal{E}'}\delta\overleftarrow{a}\,\overline{\overleftarrow{a}^{12}\underline{\overleftarrow{a}}\,\overleftarrow{b}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_4}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\underline{\overrightarrow{b}}\,\overrightarrow{b}\,\overrightarrow{b}^5\overrightarrow{a}^6\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\overrightarrow{a}^{12}\lambda\overleftarrow{b}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}\,\underline{\overrightarrow{b}}\,\overrightarrow{b}^5\overrightarrow{a}^6\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\overrightarrow{a}^{12}\lambda\overleftarrow{b}}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

On the first pass $U_{6,6}$ located the symbol to be printed by using λ as a marker. On subsequent passes $U_{6,6}$ locates the symbol to be printed by locating an $\overrightarrow{a}$ or $\overrightarrow{b}$. This printing process is iterated until $U_{6,6}$ is finished printing the ETR. The completion of this process occurs on reading a δ in state u_5 which switches $U_{6,6}$'s control to u_6:

$\boldsymbol{u_4}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^5\underline{\overrightarrow{a}}\delta\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\lambda\overleftarrow{a}^6\overleftarrow{b}^7}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^5\overleftarrow{a}\,\underline{\delta}\overrightarrow{\mathcal{E}'}\delta\overrightarrow{a}\,\overline{\lambda\overleftarrow{a}^6\overleftarrow{b}^7}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_6}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^5\overleftarrow{a}\,\lambda\underline{\overrightarrow{\mathcal{E}'}}\delta\overrightarrow{a}\,\overline{\lambda\overleftarrow{a}^6\overleftarrow{b}^7}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 3 (Restore tape)

$$
\begin{array}{l}
u_6, \overrightarrow{a}, \overleftarrow{a}, R, u_6 \\
u_6, \overrightarrow{b}, \overleftarrow{b}, R, u_6 \\
u_6, \lambda, \overrightarrow{b}, R, u_5 \\
u_6, \delta, \lambda, R, u_6
\end{array}
$$

These TRs restore M's simulated tape and encoded table of behaviour. $U_{6,6}$ moves right restoring each $\overrightarrow{a}$ to an $\overleftarrow{a}$, each $\overrightarrow{b}$ to a $\overleftarrow{b}$, and each δ to a λ. This continues until $U_{6,6}$ reads a λ, switching $U_{6,6}$'s control to u_5.

$\boldsymbol{u_6}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overline{\underline{\lambda}\overleftarrow{a}\,\overleftarrow{a}^5\overleftarrow{b}^7}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overline{\overrightarrow{b}\,\underline{\overleftarrow{a}}\,\overleftarrow{a}^5\overleftarrow{b}^7}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^{\omega}$

Cycle 4 (Choose read or write symbol)

$u_5, \overleftarrow{a}, \overleftarrow{a}, L, u_1$	$u_3, \overrightarrow{b}, \overleftarrow{a}, L, u_5$	$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$
$u_5, \overleftarrow{b}, \overleftarrow{a}, L, u_3$		$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$
u_5, λ		

This cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if $U_{6,6}$ is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if $U_{6,6}$ is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state. Case (ii) follows:

$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\overline{\underline{\overrightarrow{b}}\,\overleftarrow{a}\overleftarrow{a}^5\overleftarrow{b}^7}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^\omega$

$\boldsymbol{u_1}, (\lambda\overleftarrow{\mathcal{E}})^4\lambda\overleftarrow{\mathcal{E}'}(\lambda\overleftarrow{\mathcal{E}})^3\lambda\overleftarrow{b}^7\overleftarrow{a}^6\lambda\overleftarrow{\mathcal{E}'}\lambda\overleftarrow{a}\,\overleftarrow{\boldsymbol{b}}\,\underline{\overleftarrow{\boldsymbol{a}}}\,\overline{\overleftarrow{a}^5\overleftarrow{b}^7}\overleftarrow{b}\overleftarrow{a}\overleftarrow{a}^\omega$

In the configuration immediately above we have shortened the overlined section; the two symbols to the left of $\widehat{M_2}$'s encoded current state encode the write symbol 1.

The example simulation of TR $t_{1,1} = (q_1, 1, 1, R, q_2)$ is now complete. The correct encoded write symbol $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$ has been written and the new encoded current state is of the correct form. The new encoded current state (M_2's simulated tape head) is configured so $U_{6,6}$ reads the next encoded read symbol to the right when searching for the next ETR.

Left shifting TRs are simulated in a similar fashion to the right shifting TR given above, except in this case the write symbol is written on the right hand side of the encoded current state as shown in Fig. 2 (cL). After the left shift the new current state (M_2's simulated tape head) is configured to read the next symbol to its left when searching for the next ETR.

5.2 Construction of $U_{5,7}$

For $U_{5,7}$ the start state of $\widehat{M}$ is encoded as $\widehat{q_1} = \overleftarrow{a}^{5|Q|}\overleftarrow{b}^4$. The encoding of M's current state is of the form $\overleftarrow{a}^*\overleftarrow{b}^4\overleftarrow{b}^*\{\overleftarrow{a} \cup \epsilon\}$ and is of length $5|Q| + 4$.

Let $t = (q_x, \sigma_1, \sigma_2, D, q_y)$ be a fixed TR in M, then t is encoded via $\mathcal{P}$ using the function $\mathcal{E}$ on its own or in conjunction with $\mathcal{E}'$ where

$$\mathcal{E}(t) = \begin{cases} \overleftarrow{b}^{b(t)+2}\overleftarrow{a}^{a(,t)+1} & \text{if } D = R, \sigma_2 = 0, \\ \overleftarrow{b}^{b(t)+2}\overleftarrow{a}^{a(,t)}\overleftarrow{b} & \text{if } D = R, \sigma_2 = 1, \\ \overleftarrow{a}\overleftarrow{a}\overleftarrow{a}\overleftarrow{b}^{b(t)+2}\overleftarrow{a}^{a(,t)-1} & \text{if } D = L, \sigma_2 = 0, \\ \overleftarrow{a}\overleftarrow{b}\overleftarrow{a}\overleftarrow{b}^{b(t)+2}\overleftarrow{a}^{a(,t)-1} & \text{if } D = L, \sigma_2 = 1, \end{cases} \tag{12}$$

and

$$\mathcal{E}'(f, t) = \begin{cases} \overleftarrow{b}^{b(t^{R,x})+4}\overleftarrow{a}^{a(,t^{R,x})-2} & \text{if } \exists t^{R,x}, q_x \neq q_1, \\ \overleftarrow{b}^6\overleftarrow{a}^{5|Q|-2} & \text{if } q_x = q_1, \\ \epsilon & \text{if } \neg\exists t^{R,x}, q_x \neq q_1, \end{cases} \tag{13}$$

where as before $t^{R,x}$ is any right shifting TR such that $t^{R,x} \vdash t$.

The value of E from Definition 2 for $U_{5,7}$ is $E = \lambda \overleftarrow{b} \lambda \overleftarrow{a}$.

Definition 8 **($U_{5,7}$).** *The TM $U_{5,7}$ is defined as $U_{5,7} = (\{u_1, u_2, u_3, u_4, u_5\}, \{\overleftarrow{a}, \overleftarrow{b}, \overrightarrow{a}, \overrightarrow{b}, \lambda, \overleftarrow{\lambda}, \overrightarrow{\lambda}\}, \overleftarrow{a}, f, u_1, \{u_4, u_5\})$ where f is given by the following transition rules.*

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overrightarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \overrightarrow{b}, L, u_3$
$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overrightarrow{a}, \overleftarrow{a}, R, u_2$	$u_3, \overrightarrow{a}, \overrightarrow{a}, R, u_1$
$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$	$u_2, \overrightarrow{b}, \overleftarrow{b}, R, u_2$	$u_3, \overrightarrow{b}, \overrightarrow{b}, R, u_2$
$u_1, \lambda, \overleftarrow{a}, L, u_2$	$u_2, \lambda, \overleftarrow{b}, L, u_2$	$u_3, \lambda, \overleftarrow{\lambda}, R, u_1$
$u_1, \overleftarrow{\lambda}, \overleftarrow{b}, R, u_5$	$u_2, \overleftarrow{\lambda}, \overleftarrow{a}, L, u_3$	$u_3, \overleftarrow{\lambda}, \overrightarrow{\lambda}, L, u_3$
$u_1, \overrightarrow{\lambda}, \overleftarrow{\lambda}, R, u_1$	$u_2, \overrightarrow{\lambda}, \overleftarrow{\lambda}, R, u_2$	$u_3, \overrightarrow{\lambda}, \lambda, R, u_5$

$u_4, \overleftarrow{a}, \overrightarrow{a}, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, L, u_5$
$u_4, \overleftarrow{b}, \overrightarrow{b}, L, u_4$	$u_5, \overleftarrow{b}, \overleftarrow{a}, R, u_1$
$u_4, \overrightarrow{a}, \overleftarrow{a}, R, u_3$	$u_5, \overrightarrow{a}, \overleftarrow{a}, R, u_5$
$u_4, \overrightarrow{b}, \overleftarrow{b}, R, u_3$	$u_5, \overrightarrow{b}, \overleftarrow{b}, R, u_5$
$u_4, \lambda, \lambda, R, u_3$	$u_5, \lambda, \overleftarrow{\lambda}, L, u_1$
$u_4, \overleftarrow{\lambda}, \overrightarrow{\lambda}, L, u_4$	$u_5, \overleftarrow{\lambda},$
$u_4, \overrightarrow{\lambda},$	$u_5, \overrightarrow{\lambda}, \lambda, R, u_5$

Remark 4. There are some minor differences between the operation of $U_{5,7}$ and $U_{3,11}$. The order of symbols in ETRs of $U_{5,7}$ is reversed when compared with ETRs of $U_{3,11}$, assuming $\overleftarrow{a} = e$ and $\overleftarrow{b} = h$. To see this note the difference between Equations (3) and (12). When printing an ETR, $U_{5,7}$ reverses the order so that encoded current states are of the same form as those in $U_{3,11}$. Also M's encoded tape symbols to the left and right of the encoded current state use the *same* encodings ($\widehat{0} = \overleftarrow{a}\overleftarrow{a}$ and $\widehat{1} = \overleftarrow{b}\overleftarrow{a}$). This is not the case for $U_{3,11}$.

We give a brief overview of $U_{5,7}$ computation

Cycle 1 (Index next ETR)

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overrightarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \overrightarrow{b}, L, u_3$
$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$		$u_3, \lambda, \overleftarrow{\lambda}, R, u_1$
$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$		$u_3, \overleftarrow{\lambda}, \overrightarrow{\lambda}, L, u_3$
$u_1, \overrightarrow{\lambda}, \overleftarrow{\lambda}, R, u_1$		

In Cycle 1 the leftmost block of TRs (above) reads the encoded current state. The rightmost block neutralises λ markers by changing them to $\overleftarrow{\lambda}$ or $\overrightarrow{\lambda}$ to index the next ETR. The middle block decides when the cycle is complete. Each $\overleftarrow{b}$ in the encoded current state is changed to an $\overleftarrow{a}$ and then $U_{5,7}$ enters state u_3 via u_2.

Cycle 2 (Print ETR)

$u_4, \overleftarrow{a}, \overrightarrow{a}, L, u_4$	$u_3, \overrightarrow{a}, \overrightarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_1, \overrightarrow{a}, \overleftarrow{a}, R, u_1$
$u_4, \overleftarrow{b}, \overrightarrow{b}, L, u_4$	$u_3, \overrightarrow{b}, \overrightarrow{b}, R, u_2$	$u_2, \overrightarrow{a}, \overleftarrow{a}, R, u_2$	$u_1, \overrightarrow{b}, \overleftarrow{b}, R, u_1$
$u_4, \overrightarrow{a}, \overleftarrow{a}, R, u_3$	$u_3, \overrightarrow{\lambda}, \lambda, R, u_5$	$u_2, \overrightarrow{b}, \overleftarrow{b}, R, u_2$	$u_1, \lambda, \overleftarrow{a}, L, u_2$
$u_4, \overrightarrow{b}, \overleftarrow{b}, R, u_3$		$u_2, \lambda, \overleftarrow{b}, L, u_2$	$u_1, \overrightarrow{\lambda}, \overleftarrow{\lambda}, R, u_1$
$u_4, \lambda, \lambda, R, u_3$		$u_2, \overrightarrow{\lambda}, \overleftarrow{\lambda}, R, u_2$	
$u_4, \overleftarrow{\lambda}, \overrightarrow{\lambda}, L, u_4$			

This cycle copies an ETR to M's simulated tape head position. The leftmost block scans left and locates the next symbol of the ETR to be printed. The second block from the left records the symbol to be printed or ends the cycle. The rightmost two blocks scan right and print the appropriate symbol.

Cycle 3 (Restore tape)

$$\begin{array}{l} u_5, \overrightarrow{a}, \overleftarrow{a}, R, u_5 \\ u_5, \overrightarrow{b}, \overleftarrow{b}, R, u_5 \\ u_5, \lambda, \overleftarrow{\lambda}, L, u_1 \\ u_5, \overrightarrow{\lambda}, \lambda, R, u_5 \end{array}$$

These TRs restore M's simulated tape and encoded table of behaviour. $U_{5,7}$ moves right restoring each $\overrightarrow{a}$ to an $\overleftarrow{a}$, each $\overrightarrow{b}$ to a $\overleftarrow{b}$, and each $\overrightarrow{\lambda}$ to a λ. This continues until $U_{5,7}$ reads a λ, switching $U_{5,7}$'s control to u_1.

Cycle 4 (Choose read or write symbol)

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{\lambda}, \overleftarrow{a}, L, u_3$	$u_5, \overleftarrow{a}, \overleftarrow{a}, L, u_5$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$		$u_5, \overleftarrow{b}, \overleftarrow{a}, R, u_1$
$u_1, \overleftarrow{\lambda}, \overleftarrow{b}, R, u_5$		

This Cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if $U_{5,7}$ is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if $U_{5,7}$ is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state.

The halting case for $U_{5,7}$ is more complex than the previous UTMs. If the simulated tape head is attempting to left shift at the left end of the simulated tape then $U_{5,7}$ has the following configuration:

$$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}'})^2\lambda\lambda\overleftarrow{b}\lambda\underline{\lambda}\overline{\overrightarrow{a}\overrightarrow{a}^*\overrightarrow{b}^4\overrightarrow{b}^*}(\overrightarrow{b}\overrightarrow{a}\cup\overrightarrow{a}\overrightarrow{a})^*\overleftarrow{a}^\omega$$

$U_{5,7}$ goes through 13 configurations before the halting configuration given below is reached.

$$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}'})^2\lambda\lambda\overleftarrow{a}\overleftarrow{b}\underline{\overleftarrow{\lambda}}\overline{\overrightarrow{a}\overrightarrow{a}^*\overrightarrow{b}^4\overrightarrow{b}^*}(\overrightarrow{b}\overrightarrow{a}\cup\overrightarrow{a}\overrightarrow{a})^*\overleftarrow{a}^\omega$$

There is no TR for 'when in state u_5 read a $\overleftarrow{\lambda}$' in $U_{5,7}$ so the simulation halts.

5.3 Construction of $U_{7,5}$

For $U_{7,5}$ the start state of $\widehat{M}$ is encoded as $\widehat{q_1} = \overleftarrow{a}^{5|Q|+1}\overleftarrow{b}^3$. The encoding of M's current state is of the form $\overleftarrow{a}^*\overrightarrow{b}^3\overleftarrow{b}^*\{\overleftarrow{a}\cup\epsilon\}$ and is of length $5|Q|+4$.

Let $t = (q_x, \sigma_1, \sigma_2, D, q_y)$ be a fixed TR in M, then t is encoded via $\mathcal{P}$ using the function $\mathcal{E}$ on its own or in conjunction with $\mathcal{E}'$ where

$$\mathcal{E}(t) = \begin{cases} \overleftarrow{b}^{\,b(t)+1}(\overleftarrow{a}\,\overleftarrow{b})^{a(,t)+1}\overleftarrow{b} & \text{if } D = R, \sigma_2 = 0, \\ \overleftarrow{b}^{\,b(t)+1}(\overleftarrow{a}\,\overleftarrow{b})^{a(,t)-1}\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{b} & \text{if } D = R, \sigma_2 = 1, \\ (\overleftarrow{a}\,\overleftarrow{b})^3\,\overleftarrow{b}^{\,b(t)+1}(\overleftarrow{a}\,\overleftarrow{b})^{a(,t)-1}\overleftarrow{b} & \text{if } D = L, \sigma_2 = 0, \\ \overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{b}^{\,b(t)+1}(\overleftarrow{a}\,\overleftarrow{b})^{a(,t)-1}\overleftarrow{b} & \text{if } D = L, \sigma_2 = 1, \end{cases} \tag{14}$$

and

$$\mathcal{E}'(f, t) = \begin{cases} \overleftarrow{b}^{\,b(t^{R,x})+3}(\overleftarrow{a}\,\overleftarrow{b})^{a(,t^{R,x})-2}\overleftarrow{b} & \text{if } \exists t^{R,x}, q_x \neq q_1, \\ \overleftarrow{b}^{\,5}(\overleftarrow{a}\,\overleftarrow{b})^{5|Q|-2}\overleftarrow{b} & \text{if } q_x = q_1, \\ \epsilon & \text{if } \neg\exists t^{R,x}, q_x \neq q_1, \end{cases} \tag{15}$$

where as before $t^{R,x}$ is any right shifting TR such that $t^{R,x} \vdash t$.

The value of E from Definition 2 for $U_{7,5}$ is $E = \overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}\,\lambda\overleftarrow{a}$.

Definition 9 **($U_{7,5}$).** *The TM $U_{7,5}$ is defined as $U_{7,5} = (\{u_1, u_2, u_3, u_4, u_5, u_6, u_7\}, \{\overleftarrow{a}, \overleftarrow{b}, \lambda, \delta, \gamma\}, \overleftarrow{a}, f, u_1, \{u_2, u_5\})$ where f is given by the following transition rules.*

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \gamma, L, u_4$	$u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \lambda, L, u_3$
$u_1, \lambda, \overleftarrow{b}, R, u_1$	$u_2, \lambda, \gamma, R, u_1$	$u_3, \lambda, \delta, R, u_1$
$u_1, \delta, \delta, R, u_1$	$u_2, \delta,$	$u_3, \delta, \delta, L, u_3$
$u_1, \gamma, \overleftarrow{a}, L, u_2$	$u_2, \gamma, \overleftarrow{b}, R, u_6$	$u_3, \gamma, \overleftarrow{a}, R, u_5$
$u_4, \overleftarrow{a}, \overleftarrow{a}, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, R, u_2$	$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_6$
$u_4, \overleftarrow{b}, \lambda, L, u_4$	$u_5, \overleftarrow{b}, \overleftarrow{a}, R, u_3$	$u_6, \overleftarrow{b}, \overleftarrow{a}, L, u_7$
$u_4, \lambda, \lambda, R, u_5$	$u_5, \lambda, \gamma, R, u_6$	$u_6, \lambda, \overleftarrow{b}, R, u_6$
$u_4, \delta, \delta, L, u_4$	$u_5, \delta, \lambda, R, u_7$	$u_6, \delta, \delta, R, u_6$
$u_4, \gamma, \overleftarrow{b}, R, u_5$	$u_5, \gamma,$	$u_6, \gamma, \overleftarrow{b}, L, u_2$
$u_7, \overleftarrow{a}, \overleftarrow{a}, R, u_7$		
$u_7, \overleftarrow{b}, \overleftarrow{a}, R, u_1$		
$u_7, \lambda, \overleftarrow{b}, R, u_7$		
$u_7, \delta, \lambda, R, u_7$		
$u_7, \gamma, \gamma, L, u_5$		

Remark 5. There are some minor differences between the operation of $U_{7,5}$ and $U_{3,11}$. The order of symbols in ETRs of $U_{7,5}$ is reversed when compared with ETRs of $U_{3,11}$, assuming $\overleftarrow{a}\,\overleftarrow{b} = e$ and $\overleftarrow{b} = h$. To see this note the difference between Equations (3) and (14). When printing an ETR, $U_{7,5}$ reverses the order so that encoded current states are of the same form as $U_{3,11}$. Also M's encoded tape symbols to the left and right of the encoded current state use the *same* encodings ($\widehat{0} = \overleftarrow{a}\,\overleftarrow{a}$ and $\widehat{1} = \overleftarrow{b}\,\overleftarrow{a}$). This is not the case for $U_{3,11}$.

We give a brief overview of $U_{7,5}$'s computation.

Cycle 1 (Index next ETR)

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \gamma, L, u_4$	$u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \lambda, L, u_3$
$u_1, \lambda, \overleftarrow{b}, R, u_1$		$u_3, \lambda, \delta, R, u_1$
$u_1, \delta, \delta, R, u_1$		$u_3, \delta, \delta, L, u_3$

In Cycle 1 the leftmost block of TRs (above) reads the encoded current state. The rightmost block neutralises λ markers by changing them to δ symbols to index the next ETR. The middle block decides when the cycle is complete. Each $\overleftarrow{b}$ in the encoded current state is changed to an $\overleftarrow{a}$ and $U_{7,5}$ then enters state u_3 via u_2.

Cycle 2 (Print ETR)

$u_2, \overleftarrow{a}, \gamma, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, R, u_2$	$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_6$	$u_2, \lambda, \gamma, R, u_1$
$u_4, \overleftarrow{a}, \overleftarrow{a}, L, u_4$	$u_5, \lambda, \gamma, R, u_6$	$u_6, \lambda, \overleftarrow{b}, R, u_6$	$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$
$u_4, \overleftarrow{b}, \lambda, L, u_4$	$u_5, \delta, \lambda, R, u_7$	$u_6, \delta, \delta, R, u_6$	$u_1, \lambda, \overleftarrow{b}, R, u_1$
$u_4, \lambda, \lambda, R, u_5$		$u_6, \gamma, \overleftarrow{b}, L, u_2$	$u_1, \delta, \delta, R, u_1$
$u_4, \delta, \delta, L, u_4$			$u_1, \gamma, \overleftarrow{a}, L, u_2$
$u_4, \gamma, \overleftarrow{b}, R, u_5$			

This cycle copies an ETR to M's simulated tape head position. The leftmost block scans left and locates the next symbol of the ETR to be printed. The second block from the left records the symbol to be printed or ends the cycle. The rightmost two blocks scan right and print the appropriate symbol.

Cycle 3 (Restore tape)

$$
\begin{array}{l}
u_7, \overleftarrow{a}, \overleftarrow{a}, R, u_7 \\
u_7, \lambda, \overleftarrow{b}, R, u_7 \\
u_7, \delta, \lambda, R, u_7 \\
u_7, \gamma, \gamma, L, u_5
\end{array}
$$

These TRs restore M's simulated tape and encoded table of behaviour. $U_{7,5}$ moves right restoring each λ to a $\overleftarrow{b}$, and each δ to a λ. This continues until $U_{7,5}$ reads a γ, switching $U_{7,5}$'s control to u_5.

Cycle 4 (Choose read or write symbol)

$u_5, \overleftarrow{a}, \overleftarrow{a}, R, u_2$	$u_2, \gamma, \overleftarrow{b}, R, u_6$	$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_6$	$u_7, \overleftarrow{a}, \overleftarrow{a}, R, u_7$
$u_5, \overleftarrow{b}, \overleftarrow{a}, R, u_3$	$u_3, \gamma, \overleftarrow{a}, R, u_5$	$u_6, \overleftarrow{b}, \overleftarrow{a}, L, u_7$	$u_7, \overleftarrow{b}, \overleftarrow{a}, R, u_1$

This Cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if $U_{7,5}$ is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if $U_{7,5}$ is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state.

The halting case for $U_{7,5}$ is more complex than the first two UTMs in this paper. When the simulated tape head is attempting to left shift at the left end of the simulated tape then $U_{5,7}$ has the following configuration:

$\boldsymbol{u_7}, (\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}'})^2\lambda\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}\,\lambda\underline{\gamma}\overline{\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a}^*\,\overleftarrow{b}^3\,\overleftarrow{b}^*}(\overrightarrow{b}\,\overrightarrow{a}\cup\overrightarrow{a}\,\overrightarrow{a})^*\,\overleftarrow{a}^\omega$

$U_{5,7}$ goes through 42 configurations before the halting configuration given below is reached.

$\boldsymbol{u_5}, (\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}}\lambda\overleftarrow{\mathcal{E}'})^2\lambda\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}\,\underline{\gamma}\,\overleftarrow{b}\,\overline{\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}^*\,\overleftarrow{b}^3\,\overleftarrow{b}^*}(\overrightarrow{b}\,\overrightarrow{a}\cup\overrightarrow{a}\,\overrightarrow{a})^*\,\overleftarrow{a}^\omega$

There is no TR for 'when in state u_5 read a γ' in $U_{7,5}$ so the simulation halts.

5.4 Construction of $U_{8,4}$

For $U_{8,4}$ the start state of $\widehat{M}$ is encoded as $\widehat{q_1} = \overleftarrow{a}^{5|Q|}\,\overleftarrow{b}^2$. The encoding of M's current state is of the form $\overleftarrow{a}^*\,\overleftarrow{b}^2\,\overleftarrow{b}^*\{\overleftarrow{a}\cup\epsilon\}$ and is of length $5|Q|+2$.

Let $t = (q_x, \sigma_1, \sigma_2, D, q_y)$ be a fixed TR in M, then t is encoded via $\mathcal{P}$ using the function $\mathcal{E}$ on its own or in conjunction with $\mathcal{E}'$ where

$$\mathcal{E}(t)=\begin{cases} \overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{a(,t)}\,\overleftarrow{b}^{2(b(t))}\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } D=R, \sigma_2=0,\\ \overleftarrow{a}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{a(,t)-1}\,\overleftarrow{b}^{2(b(t))}\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } D=R, \sigma_2=1,\\ \overleftarrow{a}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{a(,t)-1}\,\overleftarrow{b}^{2(b(t))}(\overleftarrow{a}\,\overleftarrow{b}\,)^3\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } D=L, \sigma_2=0,\\ \overleftarrow{a}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{a(,t)-1}\,\overleftarrow{b}^{2(b(t))}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{b}\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } D=L, \sigma_2=1, \end{cases} \tag{16}$$

and

$$\mathcal{E}'(f,t)=\begin{cases} \overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{a(,t^{R,x})-3}\,\overleftarrow{b}^{2(b(t^{R,x})+2)}\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } \exists t^{R,x}, q_x\neq q_1,\\ \overleftarrow{b}\,\overleftarrow{b}\,\overleftarrow{a}\,(\overleftarrow{a}\,\overleftarrow{b}\,)^{5|Q|-3}\,\overleftarrow{b}^{8}\,\overleftarrow{a}\,\overleftarrow{a} & \text{if } q_x=q_1,\\ \overleftarrow{a} & \text{if } \neg\exists t^{R,x}, q_x\neq q_1, \end{cases} \tag{17}$$

where as before $t^{R,x}$ is any right shifting TR such that $t^{R,x}\vdash t$.

The value of E from Definition 2 for $U_{8,4}$ is $E=\epsilon$.

Definition 10 **($U_{8,4}$).** *The TM $U_{8,4}$ is defined as $U_{8,4} = (\{u_1,u_2,u_3,u_4,u_5,u_6,u_7,u_8\}, \{\overleftarrow{a}, \overleftarrow{b}, \lambda, \delta\}, \overleftarrow{a}, f, u_1, \{u_2\})$ where f is given by the following transition rules.*

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \delta, L, u_3$
$u_1, \lambda, \overleftarrow{b}, L, u_2$	$u_2, \lambda,$	$u_3, \lambda, \delta, R, u_1$
$u_1, \delta, \delta, R, u_1$	$u_2, \delta,$	$u_3, \delta, \delta, L, u_3$
$u_4, \overleftarrow{a}, \overleftarrow{a}, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, R, u_5$	$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_7$
$u_4, \overleftarrow{b}, \delta, L, u_5$	$u_5, \overleftarrow{b}, \delta, R, u_1$	$u_6, \overleftarrow{b}, \overleftarrow{a}, L, u_7$
$u_4, \lambda, \lambda, R, u_6$	$u_5, \lambda, \overleftarrow{a}, L, u_2$	$u_6, \lambda, \overleftarrow{b}, R, u_6$
$u_4, \delta, \delta, L, u_4$	$u_5, \delta, \delta, R, u_5$	$u_6, \delta, \overleftarrow{b}, R, u_8$
$u_7, \overleftarrow{a}, \overleftarrow{a}, R, u_6$	$u_8, \overleftarrow{a}, \overleftarrow{a}, R, u_6$	
$u_7, \overleftarrow{b}, \overleftarrow{a}, R, u_1$	$u_8, \overleftarrow{b}, \overleftarrow{a}, L, u_3$	
$u_7, \lambda, \overleftarrow{a}, R, u_1$	$u_8, \lambda, \overleftarrow{a}, L, u_8$	
$u_7, \delta, \lambda, R, u_6$	$u_8, \delta, \overleftarrow{b}, R, u_6$	

We give a brief overview of $U_{8,4}$'s computation. The tape contents is given by the same symbols ($\widehat{1} = \overleftarrow{b}\,\overleftarrow{a}$ and $\widehat{0} = \overleftarrow{a}\,\overleftarrow{a}$) to the left and right of the simulated TMs tape head.

Cycle 1 (Index next ETR)

$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$	$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_3, \overleftarrow{a}, \overleftarrow{a}, L, u_3$
$u_1, \overleftarrow{b}, \overleftarrow{a}, R, u_2$	$u_2, \overleftarrow{b}, \overleftarrow{b}, L, u_3$	$u_3, \overleftarrow{b}, \delta, L, u_3$
$u_1, \delta, \delta, R, u_1$		$u_3, \lambda, \delta, R, u_1$
		$u_3, \delta, \delta, L, u_3$

In Cycle 1 the leftmost block of TRs (above) reads the encoded current state. The rightmost block neutralises markers to index the next ETR. The middle block decides when the cycle is complete. In u_1 $U_{8,4}$ scans the encoded current state from left to right; each $\overleftarrow{b}$ is changed to an $\overleftarrow{a}$ and $U_{8,4}$ then enters state u_3 via u_2.

Cycle 2 (Print ETR)

$u_2, \overleftarrow{a}, \lambda, L, u_4$	$u_5, \overleftarrow{a}, \overleftarrow{a}, R, u_5$	$u_1, \overleftarrow{a}, \overleftarrow{a}, R, u_1$
$u_4, \overleftarrow{a}, \overleftarrow{a}, L, u_4$	$u_5, \overleftarrow{b}, \delta, R, u_1$	$u_1, \lambda, \overleftarrow{b}, L, u_2$
$u_4, \overleftarrow{b}, \delta, L, u_5$	$u_5, \lambda, \overleftarrow{a}, L, u_2$	$u_1, \delta, \delta, R, u_1$
$u_4, \lambda, \lambda, R, u_6$	$u_5, \delta, \delta, R, u_5$	
$u_4, \delta, \delta, L, u_4$		

Before we explain this cycle we mention why ETRs for $U_{8,4}$ are longer than ETRs for the other UTMs (e.g. compare Equations (16) and (3)). In $U_{8,4}$'s ETRs there are multiple copies of the subwords $\overleftarrow{a}\,\overleftarrow{b}$ and $\overleftarrow{b}\,\overleftarrow{b}$. During the Print ETR cycle, the subword $\overleftarrow{a}\,\overleftarrow{b}$ will cause an $\overleftarrow{a}$ to be printed and the subword $\overleftarrow{b}\,\overleftarrow{b}$ will cause a $\overleftarrow{b}$ to be printed. During this cycle the next symbol to be printed is the symbol to the left of the rightmost $\overleftarrow{b}$ in the ETR. The rightmost $\overleftarrow{b}$ of the subwords $\overleftarrow{a}\,\overleftarrow{b}$ and $\overleftarrow{b}\,\overleftarrow{b}$ is simply a marker and the symbol directly to its left is the symbol that is to be printed. Extra $\overleftarrow{a}$ symbols appear in $U_{8,4}$'s ETRs that do not result in symbols being printed during the print ETR cycle. These extra $\overleftarrow{a}$ symbols are added to allow the restore tape cycle to execute correctly.

This cycle copies an ETR to M's simulated tape head position. The leftmost block scans left and locates the next symbol of the ETR to be printed or ends the cycle. The middle block records the symbol to be printed. If an $\overleftarrow{a}$ is to be printed the middle block also scans right and prints an $\overleftarrow{a}$. If a $\overleftarrow{b}$ is to be printed the rightmost block scan right and prints a $\overleftarrow{b}$.

Cycle 3 (Restore tape)

$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_7$	$u_7, \overleftarrow{a}, \overleftarrow{a}, R, u_6$	$u_8, \overleftarrow{a}, \overleftarrow{a}, R, u_6$
$u_6, \lambda, \overleftarrow{b}, R, u_6$	$u_7, \lambda, \overleftarrow{a}, R, u_1$	$u_8, \lambda, \overleftarrow{a}, L, u_8$
$u_6, \delta, \overleftarrow{b}, R, u_8$	$u_7, \delta, \lambda, R, u_6$	$u_8, \delta, \overleftarrow{b}, R, u_6$

These TRs restore M's simulated tape and encoded table of behaviour. $U_{8,4}$'s tape head scans right restoring δ symbols to $\overleftarrow{b}$ and λ symbols. Recall that in the Index next ETR cycle λ symbols were change to δ symbols in order to index the next ETR. Note also that during the Index next ETR cycle as $U_{8,4}$ scans left it also changes $\overleftarrow{b}$ symbols to

δ symbols. As mentioned earlier there are extra $\overleftarrow{a}$ symbols in each ETR that do not effect what is printed to the overlined region. The reason for these extra $\overleftarrow{a}$ symbols is to ensure that $U_{8,4}$ can distinguish which δ symbols to restore to λ symbols and which δ symbols to restore to $\overleftarrow{b}$ symbols. The extra $\overleftarrow{a}$ symbols ensure that $U_{8,4}$ will be in state u_7 if a δ should be restored to a λ and in u_6 or u_8 if a δ should be restored to a $\overleftarrow{b}$. This cycle ends when $U_{8,4}$ reads a λ.

Cycle 4 (Choose read or write symbol)

$u_6, \overleftarrow{a}, \overleftarrow{a}, R, u_7$ $\quad$ $u_7, \overleftarrow{b}, \overleftarrow{a}, R, u_1$ $\quad$ $u_8, \overleftarrow{b}, \overleftarrow{a}, L, u_3$
$u_6, \overleftarrow{b}, \overleftarrow{a}, L, u_7$

This cycle either (i) begins the indexing of an ETR or (ii) completes the execution of an ETR. More precisely: (i) if $U_{8,4}$ is immediately after simulating a left shift then this cycle reads the encoded read symbol to the left of the encoded current state, (ii) if $U_{8,4}$ is simulating a right shift then this cycle prints the encoded write symbol to the left of the encoded current state.

Remark 6. Halting case $U_{8,4}$. Recall that all our UTMs simulate halting by attempting to simulate a left shift at the left end of the simulated tape. This is also true for $U_{8,4}$. However the halting case for $U_{8,4}$ differs slightly from the halting case for $U_{3,11}$. $U_{3,11}$ halts during the *Choose read or write symbol cycle*. $U_{8,4}$ halts in the configuration immediately after printing the last symbol of the left shifting ETR at the end of the *Print ETR cycle*.

6 Conclusion and future work

We have improved the state of the art in small efficient UTMs. Fig. 1 summarises our results. Our UTMs infer a polynomial time curve that in some places matches the already known (from Rogozhin et al) exponential time curve.

The decrease in the state-symbol product was found, in part through direct simulation of TMs. This is rather suprising given the trend over the last forty years of indirect simulation through other universal models. The most recent small UTMs simulate TMs via 2-tag systems, with an exponential time overhead [7, 14, 6, 4, 5]. Before the advent of Minsky's UTM in UTM$(7, 4)$, the smallest UTMs directly simulated TMs [2, 3]. One problemma in the construction of these UTMs was the addressing of states, that is locating the next encoded state during TR simulation. Some approaches to solving this problemma are discussed in brief in Section 3.1 of Minsky's paper [4]. A major advantage of our algorithm is the fact that the encoded current state is located at the simulated tape head position. This technique simplifies the addressing of states.

What about small UTMs with less than polynomial time complexity? For example, consider the construction of a linear time UTM. Our UTM stores the encoded current state at the simulated tape head location. Suppose the *entire* encoded table of behaviour is stored at this location. Simulating a TR involves scanning through the encoded table of behaviour, it is not necessary to scan the entire simulated tape contents. The idea is straightforward, however trying to construct *small* linear time UTMs could be difficult.

Cook [14, 15] has recently published UTMs in UTM$(2, 5)$, UTM$(3, 4)$, UTM$(4, 3)$ and UTM$(7, 2)$ that are smaller than Rogozhin et al's. However, Cook's UTMs differ from the classical Turing machine definition [13]. Instead of having a blank symbol these machines have two blank words. Cook's UTMs require the blank tape to have an infinitely repeating word to the left and another different infinitely repeating word to the right. Cooks machines also suffer from an exponential slowdown through simulation of 2-tag systems. As future work it would be interesting to find polynomial time UTMs as small as Cook's. At present it seems technically challenging to further reduce the size of our machines so we suspect that a radically different approach is required.

References

1. Shannon, C.E.: A universal Turing machine with two internal states. Automata Studies, Annals of Mathematics Studies **34** (1956) 157–165
2. Ikeno, N.: A 6-symbol 10-state universal Turing machine. In: Proceedings, Institute of Electrical Communications, Tokyo (1958)
3. Watanabe, S.: 5-symbol 8-state and 5-symbol 6-state universal Turing machines. ACM **8** (1961) 476–483
4. Minsky, M.: Size and structure of universal Turing machines using tag systems. In: Recursive Function Theory: Proceedings, Symposium in Pure Mathematics. Volume 5., Provelence, AMS (1962) 229–238
5. Rogozhin, Y.: Small universal Turing machines. Theoretical Computer Science **168** (1996) 215–240
6. Kudlek, M., Rogozhin, Y.: A universal Turing machine with 3 states and 9 symbols. In W. Kuich, G.R., Salomaa, A., eds.: Developments in Language Theory. Volume 2295 of LNCS., Vienna, DLT, Springer (2001) 311–318
7. Baiocchi, C.: Three small universal Turing machines. In Margenstern, Rogozhin, eds.: Machines, Computations and Universality. Volume 2055 of LNCS., Springer (2001) 1–10
8. Kudlek, M.: Small deterministic Turing machines. Theoretical Computer Science **168** (1996) 241–255
9. Pavlotskaya, L.: Solvability of the halting problem for certain classes of Turing machines. Mathematical Notes Academy of Science USSR **13** (1973) 537–541
10. Pavlotskaya, L.: Sufficient conditions for the halting problem decidability of Turing machines (in Russian). Avtomaty i Mashiny (1978) 91–118
11. Hermann, G.: The uniform halting problem for generalized one state Turing machines. In IEEE, ed.: Proceedings, Ninth Annual Symposium on Switching and Automata Theory, Schenectady, New York, FOCS, IEEE Computer Society Press (1968) 368–372
12. Boas, P.V.E.: 1. In: Handbook of theoretical computer science, volume A. Elsevier (1990) 1–66
13. Hopcroft, J.E., Ullman, J.D.: Introduction to automata theory, languages, and computation. Addison-Wesley (1979)
14. Cook, M.: Universality in elementary cellular automata. Complex Systems **15** (2004) 1–40
15. Wolfram, S.: A new kind of science. Wolfram Media, Inc. (2002)