Mathematica Aeterna, Vol. 7, 2017, no. 1, 19 - 56

Reverse engineering Turing Machines and insights into the Collatz conjecture.

John Nixon

Brook Cottage
The Forge, Ashburnham
Battle, East Sussex
TN33 9PH, U.K.

Abstract

In this paper I have extended my earlier work [3] on small Turing Machines (TMs) by developing a method for obtaining recursive definitions of the irreducible regular rules (IRR) for a TM when explicit formulae for them cannot be obtained. This has been illustrated by two examples. The first example was randomly chosen and the second example was designed to simulate the Collatz conjecture. Analysis of this TM based on the its IRR suggested new approaches that might be the basis for a proof of this conjecture.

The method involves running the TM backwards from a configuration set (CS). This in general produces a tree of CSs at each step. The aim is to find CS's `y` that are reachable from a CS `x` that simply specifies the symbol about to be read and the machine state. This means that following the computation forward from `x` by adding some symbols when needed at the pointer, the CS `y` can be reached. These CS's form the basis of the LHS's of the IRR.

Mathematics Subject Classification: 68Q25

Keywords: Turing machine, irreducible regular computation rules, Collatz conjecture

1 Introduction

In my earlier paper [3] I showed how a set of shortcut computation rules could be defined for any Turing Machine (TM) that can speed up calculations with it, and how any computation with the TM can be expressed more briefly in terms of these rules. By appropriately restricting the set of rules to the Irreducible Regular Rules (IRR), this desirable property is maintained and redundancy is eliminated. The number of IRR for a TM can be finite or infinite. In the examples studied in detail where they were infinite in number, general formulae for them were proved by induction. Powerful general results were then obtained for the general behaviours of the TMs studied by using their sets of IRRs.

In this paper, that closely follows on from [3], more complex examples are studied. Due to the complexity of the work, only two TM's were chosen. The TM's used are one randomly chosen (TM1) and the other (TM2) a TM which is a trivial modification of one defined by Margenstern [2] to generate sequences to which the Collatz conjecture [1] applies.

As a result of the increased complexity, the IRR cannot be expressed by simple formulae in these examples, and the methods needed to generate a recursive description of the set of IRR became much more complex, but the same principle seems to apply, namely that finding this recursive description of the IRR for the TM allows many questions about its general behaviour to be answered. The recursive descriptions in these examples now take the more general form of an algorithm for generating from the IRR of length $\mathtt{n}$, IRR($\mathtt{n}$), the set IRR($\mathtt{n}+\mathtt{1}$). This is possible for TM1 because, although the number of such rules increases with $\mathtt{n}$, the number of patterns in which they fall does not, and all the rules in a pattern can be handled by the same reasoning.

For TM1, the patterns are determined by a few symbols near the pointer in the origin for the LHS of the IRR, where the origin of a configuration set (CS) $\mathtt{x}$ is the set of CS's obtained by tracing back the computation from $\mathtt{x}$ as far as possible. The finding of the origins of a CS played an essential role in the analysis of TM2 too, and is therefore a major theme of this paper and is based on running the TM backwards.

The analysis especially for TM1 has proved to be tricky, and the reasoning used in this paper is a much simplified version of the original reasoning I used. The inductive hypotheses were arrived at with the aid of the computer program [4] I developed.

After the complete set of IRR were characterised, the analysis of TM2 provided a very plausible proposition equivalent to the well-known Collatz conjecture, and it seems that a proof of it might be found based on this result, or at least new insights into it might arise from similar analyses of TM's that simulate it.

The notation for the TM's is as follows: symbols will be small Latin letters (possibly capital etc. if needed in future), and states will be integers. Symbols which are place-holders for TM symbols will be Greek (here α and β,γ and δ). This is in contrast to the notation in [3] and [4] in which the symbols were digits and the states were capital letters. The definition of TM's used in this paper will also be very slightly different from that in [3] in that halting is treated differently. Halting (if used) will no longer be associated with a machine state H but with some lines in the machine table with no movement of the read-write head and the same state and symbol as the input to the instruction appearing in its output, so the TM is in an infinite stationary loop i.e. effectively halting. Halting instructions will then be naturally classified with other stationary cycling rules with more than one TM step.

There are many advantages for the change of notation which outweigh the inconvenience of changing it:

(1) The symbols are really just symbols not digits which have another meaning as integers.

(2) expressions involving superscripts cannot be misinterpreted as numerical expressions.

(3) The number of symbols available has increased from 10 to 26, and more if capitals etc. are used.

(4) The number of machine states representable is now unlimited.

(5) These advantages apply to new versions including the current version[5] of the computer program TIE for analysis of Turing machines.

This program also uses the programming language D (dlang.org) instead of C++ that also has other advantages.

Apart from these changes, the notation and terminology used in this paper is the same as that used in [3].

The paper is divided into 2 parts giving the analyses of Turing Machines TM1 and TM2 each according to the above outline, i.e. (1) introducing the TM and giving the IRR($\mathtt{n}$) for a few small values of $\mathtt{n}$ followed by (2) approaches to the recursive definition of the IRR, and finally (3) some general statements about the behaviour of the TM following from the results in (2). Note that in step (2) there is no attempt to make this algorithmic i.e. the form of the recursion has to be correctly guessed before it can be proved by induction. The two examples are very different so the approach in parts (2) differ greatly. For TM1 the first approach to part (2) was ultimately unsatisfactory but illustrated many ideas. This was followed by what I think is a general method based on searching for origins of CS's by running the TM backwards over all possible branches. This allowed the IRR($\mathtt{n}+\mathtt{1}$) to be obtained from the IRR($\mathtt{n}$) for $\mathtt{n} = \mathtt{1}, \mathtt{2}, \mathtt{3}$ and then for $\mathtt{n} \geq \mathtt{4}$. For TM2, this approach was followed but the results could be expressed in terms of sub-machines of the TM, so the arguments look very different. This is presumably the result of the TM being designed for a specific task. Finally I tried to exploit the connection of TM2 with the Collatz problem to shed some light on it.

2 Definition of TM1 and some preliminary results for the IRR of length $\mathtt{n} \le 4$

The TM studied in this section (TM1) is defined by its computation rules of length 1 which are as follows:

$$\begin{array}{llll}
\mathtt{1\underline{a} \to 2c_} & \mathtt{1\underline{b} \to 4_b} & \mathtt{1\underline{c} \to 1d_} & \mathtt{1\underline{d} \to 2d_} \\
\mathtt{2\underline{a} \to 3c_} & \mathtt{2\underline{b} \to 1_d} & \mathtt{2\underline{c} \to 4a_} & \mathtt{2\underline{d} \to 2\underline{d}} \\
\mathtt{3\underline{a} \to 3b_} & \mathtt{3\underline{b} \to 4_a} & \mathtt{3\underline{c} \to 3d_} & \mathtt{3\underline{d} \to 4_c} \\
\mathtt{4\underline{a} \to 1a_} & \mathtt{4\underline{b} \to 2c_} & \mathtt{4\underline{c} \to 3_b} & \mathtt{4\underline{d} \to 3c_}
\end{array}. \tag{1}$$

The complete list IRR's of lengths 2, 3 and 4 are all listed here in equations 2-4 respectively. The order in which they are listed is the same as the order in which the computer program output is listed. The results are sorted by state, then by pointer position, then lexicographically (forward or reverse) such that the symbol away from the pointer changes most quickly:

$$\begin{array}{lllll}
\mathtt{1\underline{a}d \to 2c\underline{d}} & \mathtt{1\underline{c}d \to 2dd_} & \mathtt{1\underline{d}d \to 2d\underline{d}} & \mathtt{1a\underline{b} \to 1a\underline{b}} & \mathtt{1d\underline{b} \to 3_ba} \\
\mathtt{2c\underline{b} \to 2dd_} & \mathtt{2d\underline{b} \to 2d\underline{d}} & \mathtt{3\underline{a}b \to 3cc_} & \mathtt{3\underline{c}b \to 3cb_} & \mathtt{3b\underline{b} \to 3cc_} \\
\mathtt{3c\underline{b} \to 3_ba} & \mathtt{3d\underline{b} \to 3cb_} & \mathtt{3b\underline{d} \to 4ca_} & \mathtt{3c\underline{d} \to 3_bc} & \mathtt{3d\underline{d} \to 3cd_} \\
\mathtt{4\underline{a}a \to 2ac_} & \mathtt{4\underline{a}b \to 4\underline{a}b} & \mathtt{4\underline{a}c \to 1ad_} & \mathtt{4\underline{b}a \to 3cc_} & \mathtt{4\underline{b}b \to 2dd_} \\
\mathtt{4\underline{b}c \to 4ca_} & \mathtt{4\underline{d}a \to 3cb_} & \mathtt{4\underline{d}b \to 3_ba} & \mathtt{4\underline{d}c \to 3cd_} & \mathtt{4a\underline{c} \to 3cc_}
\end{array} \tag{2}$$

$$\begin{array}{llll}
\mathtt{1ad\underline{b} \to 3ccb_} & \mathtt{1dd\underline{b} \to 4_cba} & \mathtt{3\underline{a}ba \to 3ccb_} & \mathtt{3\underline{a}bb \to 3cbb_} \\
\mathtt{3\underline{a}bc \to 3ccd_} & \mathtt{3\underline{c}ba \to 3cbb_} & \mathtt{3\underline{c}bb \to 3ccc_} & \mathtt{3\underline{c}bc \to 3cbd_} \\
\mathtt{3ac\underline{b} \to 3ccb_} & \mathtt{3cc\underline{b} \to 3cbb_} & \mathtt{3dc\underline{b} \to 4_cba} & \mathtt{3ac\underline{d} \to 3ccd_} \\
\mathtt{3cc\underline{d} \to 3cbd_} & \mathtt{3dc\underline{d} \to 4_cbc} & \mathtt{4\underline{a}ab \to 2add_} & \mathtt{4\underline{a}cb \to 3ccb_} \\
\mathtt{4\underline{b}ab \to 3cbb_} & \mathtt{4\underline{b}bd \to 2dd\underline{d}} & \mathtt{4\underline{b}cb \to 2cac_} & \mathtt{4\underline{d}ab \to 3ccc_} \\
\mathtt{4\underline{d}cb \to 3ccb_} & & &
\end{array} \tag{3}$$

$$\begin{array}{lll}
\mathtt{1add\underline{b} \to 3ccbb_} & \mathtt{1ddd\underline{b} \to 3ccbb_} & \mathtt{3\underline{a}bab \to 3cccc_} \\
\mathtt{3\underline{a}bad \to 4ccca_} & \mathtt{3\underline{a}bbd \to 4cbca_} & \mathtt{3\underline{a}bcb \to 3cccb_} \\
\mathtt{3\underline{c}bab \to 3cbcc_} & \mathtt{3\underline{c}bad \to 4cbca_} & \mathtt{3\underline{c}bbd \to 3ccbd_} \\
\mathtt{3\underline{c}bcb \to 3cbcb_} & \mathtt{3adc\underline{b} \to 3ccbb_} & \mathtt{3ddc\underline{b} \to 3ccbb_} \\
\mathtt{3adc\underline{d} \to 3ccbd_} & \mathtt{3ddc\underline{d} \to 3ccbd_} & \mathtt{4\underline{a}aba \to 3addc_} \\
\mathtt{4\underline{a}abb \to 2add\underline{d}} & \mathtt{4\underline{a}abc \to 4adda_} & \mathtt{4\underline{a}cba \to 3ccbb_} \\
\mathtt{4\underline{a}cbb \to 3cccc_} & \mathtt{4\underline{a}cbc \to 3ccbd_} & \mathtt{4\underline{b}aba \to 3cbbb_} \\
\mathtt{4\underline{b}abb \to 3cbcc_} & \mathtt{4\underline{b}abc \to 3cbbd_} & \mathtt{4\underline{b}cba \to 3cacc_} \\
\mathtt{4\underline{b}cbb \to 2cadd_} & \mathtt{4\underline{b}cbc \to 4caca_} & \mathtt{4\underline{d}aba \to 3cccb_} \\
\mathtt{4\underline{d}abb \to 3ccbb_} & \mathtt{4\underline{d}abc \to 3cccd_} & \mathtt{4\underline{d}cba \to 3ccbb_} \\
\mathtt{4\underline{d}cbb \to 3cccc_} & \mathtt{4\underline{d}cbc \to 3ccbd_} &
\end{array} \tag{4}$$

From Theorem 5.4 in [3] any rule in $\mathtt{RR}(\mathtt{n}+\mathtt{1})$, the set of regular rules of length $\mathtt{n}+\mathtt{1}$, is derived by a sequence (of length $\ge \mathtt{0}$) of rules in $\mathtt{RR}(\mathtt{n})$ of types RL and LR alternating, followed by a regular rule of type RR or LL, with matching R or L symbols as described in [3]. The same therefore applies to the subset $\mathtt{IRR}(\mathtt{n}+\mathtt{1})$ except that the number of derivation steps must be at least 2 to ensure that the derived rule is irreducible (i.e. no redundant symbols). The type of the IRR derived of length $\mathtt{n}+\mathtt{1}$ also follows from the R and L symbol matching. Therefore the absence of IRR(4) of type RL implies that the IRR(5) must all be derived in two steps by IRR of types LR followed by RR and have type LR. This argument can be clearly extended to all longer IRR. It is therefore useful to list separately in equation (5) the non-halting IRR of lengths 2-4 that are of type RR for TM1 because these must be the last rules of length $>\mathtt{1}$ used to complete the derivations of the IRR:

$$\begin{array}{llll}
\mathtt{2c\underline{b}} \to \mathtt{2dd_} & \mathtt{3b\underline{b}} \to \mathtt{3cc_} & \mathtt{3d\underline{b}} \to \mathtt{3cb_} & \mathtt{3b\underline{d}} \to \mathtt{4ca_} \\
\mathtt{4a\underline{c}} \to \mathtt{3cc_} & \mathtt{3d\underline{d}} \to \mathtt{3cd_} & \mathtt{1ad\underline{b}} \to \mathtt{3ccb_} & \mathtt{3ac\underline{b}} \to \mathtt{3ccb_} \\
\mathtt{3cc\underline{b}} \to \mathtt{3cbb_} & \mathtt{3ac\underline{d}} \to \mathtt{3ccd_} & \mathtt{3cc\underline{d}} \to \mathtt{3cbd_} & \mathtt{1add\underline{b}} \to \mathtt{3ccbb_} \\
\mathtt{1ddd\underline{b}} \to \mathtt{3ccbb_} & \mathtt{3adc\underline{b}} \to \mathtt{3ccbb_} & \mathtt{3ddc\underline{b}} \to \mathtt{3ccbb_} & \mathtt{3adc\underline{d}} \to \mathtt{3ccbd_} \\
\mathtt{3ddc\underline{d}} \to \mathtt{3ccbd_} & & &
\end{array} \tag{5}$$

2.1 Initial exploration of the irreducible regular rules for $\mathtt{n} > 4$

After looking at the IRR(4) and the table of all the derivations of the IRR(5) from these (not shown but easily obtained from the computer program [4] output), the results can be summarised in Table 1. Table 1 shows that the state of the LHS of a rule and the symbol (only one in this example) at the opposite end (i.e. the right hand end) from the pointer in this LHS determine a set of extra symbols any one of which can be added to both sides of this rule at the right hand end of the string of symbols, and in each case corresponding IRRs can be identified to finish the derivation. There could be up to 3 of these, depending on other symbols. They are listed here in an arbitrary order in the two sub-columns of the last column of Table 1.

LHS of IRR(4)		Symbol added at pointer in the RHS of the IRR(4)	IRR used to complete the derivation of the IRR(5), which depend on other symbols	
State	Symbol at opposite(R) end from pointer(L)			
3	b	a	$\mathtt{3\underline{a}} \to \mathtt{3b\_}$	
3	b	b	$\mathtt{3cc\underline{b}} \to \mathtt{3cbb\_}(1)$	$\mathtt{3b\underline{b}} \to \mathtt{3cc\_}(2)$
3	b	c	$\mathtt{3\underline{c}} \to \mathtt{3d\_}$	
4	a	b	$\mathtt{3cc\underline{b}} \to \mathtt{3cbb\_}$	$\mathtt{3b\underline{b}} \to \mathtt{3cc\_}$
			$\mathtt{3ddc\underline{b}} \to \mathtt{3ccbb\_}$	
4	a	d	$\mathtt{3cc\underline{d}} \to \mathtt{3cbd\_}$	$\mathtt{3b\underline{d}} \to \mathtt{4ca\_}$
			$\mathtt{3ddc\underline{d}} \to \mathtt{3ccbd\_}$	
4	c	b	$\mathtt{4\underline{b}} \to \mathtt{2c\_}$	$\mathtt{3d\underline{b}} \to \mathtt{3cb\_}$
4	b	d	$\mathtt{3cc\underline{d}} \to \mathtt{3cbd\_}$	$\mathtt{2\underline{d}} \to \mathtt{2\underline{d}}$
			$\mathtt{3b\underline{d}} \to \mathtt{4ca\_}$	

Table 1: Summary of the derivations of the IRR(5) from the IRR(4)

For example to explain the second line in detail, it is easy to show (e.g. with the computer program [4]) that

$$\begin{aligned} &\mathtt{3\underline{a}bab} \to \mathtt{3cccc\_} \\ &\mathtt{3\underline{a}bcb} \to \mathtt{3cccb\_} \\ &\mathtt{3\underline{c}bab} \to \mathtt{3cbcc\_} \\ &\mathtt{3\underline{c}bcb} \to \mathtt{3cbcb\_} \end{aligned}. \tag{6}$$

These are all the IRR(4) for TM1 that have the LHS with state `3` and `b` at the right hand end of the tape and the pointer is at the left. In each case a `b` can be added to both sides of these IRR on the right hand end to start the following derivations that are completed by application of one of the IRR stated in the last column of Table 1 labelled (1) or (2).

$$\begin{aligned} &\mathtt{3\underline{a}babb} \to \mathtt{3cccc\underline{b}} \to \mathtt{3cccbb\_} \quad (1) \\ &\mathtt{3\underline{a}bcbb} \to \mathtt{3cccb\underline{b}} \to \mathtt{3ccccc\_} \quad (2) \\ &\mathtt{3\underline{c}babb} \to \mathtt{3cbcc\underline{b}} \to \mathtt{3cbcbb\_} \quad (1) \\ &\mathtt{3\underline{c}bcbb} \to \mathtt{3cbcb\underline{b}} \to \mathtt{3cbccc\_} \quad (2) \end{aligned} \tag{7}$$

In line 1 of Table 1, the LHSs of the derived rules have the symbol `a`, which is the symbol added, at the opposite end from the pointer, therefore the combination `3a` must appear in the first two columns of Table 1 if it were to be applicable again to generate a member of IRR(6) because the state of the LHS of a rule `r` in IRR(`n`) is the same as the state of the LHS of the member

LHS of IRR(5)		Symbol added at pointer in the RHS of the IRR(5)	IRR used to complete the derivation of the IRR(6), which depend on other symbols	
State	Symbol at opposite(R) end from pointer(L)			
3	a	b	3b$\underline{\texttt{b}}$ → 3cc_	
3	a	d	3b$\underline{\texttt{d}}$ → 4ca_	
3	b	d	3cc$\underline{\texttt{d}}$ → 3cbd_	3b$\underline{\texttt{d}}$ → 4ca_
3	c	b	3d$\underline{\texttt{b}}$ → 3cb_	
4	b	a	3$\underline{\texttt{a}}$ → 3b_	2$\underline{\texttt{a}}$ → 3c_
4	b	b	2c$\underline{\texttt{b}}$ → 2dd_	3b$\underline{\texttt{b}}$ → 3cc_
			3cc$\underline{\texttt{b}}$ → 3cbb_	
4	b	c	2$\underline{\texttt{c}}$ → 4a_	3$\underline{\texttt{c}}$ → 3d_

Table 2: Summary of the derivations of the IRR(6) from the IRR(5)

of IRR(n + 1) derived from r. Therefore the same general way of obtaining all the IRR(6) from the IRR(5) cannot be done as in Table 1. After listing the derivations of the IRR(6) from the IRR(5) with the help of the computer program [4], the output suggests the following process (where both symbols mentioned are again added at the right hand end) for the next extension of length of the IRR by 1:

Now it looks as if an induction argument could be used, because each state and symbol-at-the-end (column 2) combination in Table 1 is found in columns 1 and 3 of Table 2 and the same holds with Tables 1 and 2 interchanged.

If the rules derived as summarised in Table 1, are used as inputs to the summary derivations in Table 2 the result is the set of rule outlines as follows:

$$
\begin{array}{llll}
\texttt{3}\ldots\texttt{bab} \to \texttt{3}\ldots\texttt{b}\underline{\texttt{b}} & \to \texttt{3}\ldots\texttt{cc\_} \\
\texttt{3}\ldots\texttt{bad} \to \texttt{3}\ldots\texttt{b}\underline{\texttt{d}} & \to \texttt{4}\ldots\texttt{ca\_} \\
\texttt{3}\ldots\texttt{bbd} \to \texttt{3}\ldots\texttt{cc}\underline{\texttt{d}} & \to \texttt{3}\ldots\texttt{cbd\_} \\
\texttt{3}\ldots\texttt{bbd} \to \texttt{3}\ldots\texttt{cbb}\underline{\texttt{d}} & \to \texttt{4}\ldots\texttt{cbca\_} \\
\texttt{3}\ldots\texttt{bcb} \to \texttt{3}\ldots\texttt{d}\underline{\texttt{b}} & \to \texttt{3}\ldots\texttt{cb\_} \\
\texttt{4}\ldots\texttt{aba} \to \texttt{3}\ldots\texttt{cc}\underline{\texttt{a}} & \to \texttt{3}\ldots\texttt{ccb\_} \\
\texttt{4}\ldots\texttt{abb} \to \texttt{3}\ldots\texttt{cc}\underline{\texttt{b}} & \to \texttt{3}\ldots\texttt{cbb\_} \\
\texttt{4}\ldots\texttt{abc} \to \texttt{3}\ldots\texttt{cc}\underline{\texttt{c}} & \to \texttt{3}\ldots\texttt{ccd\_} \\
\texttt{4}\ldots\texttt{aba} \to \texttt{3}\ldots\texttt{cbb}\underline{\texttt{a}} & \to \texttt{3}\ldots\texttt{cbbb\_}. \\
\texttt{4}\ldots\texttt{abb} \to \texttt{3}\ldots\texttt{cbb}\underline{\texttt{b}} & \to \texttt{3}\ldots\texttt{cbcc\_} \\
\texttt{4}\ldots\texttt{abc} \to \texttt{3}\ldots\texttt{cbb}\underline{\texttt{c}} & \to \texttt{3}\ldots\texttt{cbbd\_} \\
\texttt{4}\ldots\texttt{cba} \to \texttt{2}\ldots\texttt{c}\underline{\texttt{a}} & \to \texttt{3}\ldots\texttt{cc\_} \\
\texttt{4}\ldots\texttt{cbb} \to \texttt{2}\ldots\texttt{c}\underline{\texttt{b}} & \to \texttt{2}\ldots\texttt{dd\_} \\
\texttt{4}\ldots\texttt{cbc} \to \texttt{2}\ldots\texttt{c}\underline{\texttt{c}} & \to \texttt{4}\ldots\texttt{ca\_} \\
\texttt{4}\ldots\texttt{cba} \to \texttt{3}\ldots\texttt{cb}\underline{\texttt{a}} & \to \texttt{3}\ldots\texttt{cbb\_} \\
\texttt{4}\ldots\texttt{cbb} \to \texttt{3}\ldots\texttt{cb}\underline{\texttt{b}} & \to \texttt{3}\ldots\texttt{ccc\_} \\
\texttt{4}\ldots\texttt{cbc} \to \texttt{3}\ldots\texttt{cb}\underline{\texttt{c}} & \to \texttt{3}\ldots\texttt{cbd\_}
\end{array} \tag{8}
$$

For example the rules derived in the first row of Table 1 are of the form $\texttt{3}\ldots\texttt{ba} \to \texttt{3}\ldots\texttt{b\_}$ where the pointer is at the left in the LHS and $\ldots$ stands for arbitrary symbols that may be different at each instance of it. This means `3` "some string of symbols ending with `ba`" with the pointer on the left symbol leads to `3` "some other string of symbols ending with `b`" with the pointer to the immediate right of the `b`, and will be called a rule outline. Now the first two rows of Table 2 apply giving the first two rows of (8).

Note that the third rule for the fourth row in column 4 of Table 1 has its RHS `3ccbb_`, a subset of the RHS `3cbb_` of another rule for the same row of Table 1. Thus $\texttt{4}\ldots\texttt{ab} \to \texttt{3}\ldots\texttt{cbb\_}$ and $\texttt{4}\ldots\texttt{ab} \to \texttt{3}\ldots\texttt{ccbb\_}$ could both be derived from Table 1 and the latter is a special case of the former so the latter will not used in (8). There is another example of this in row 5 of Table 1.

After completing the rest of (8) in this way (noting that the appropriate rule has to be selected if more than one are given in Table 2), one can deduce that the rules of Table 1 are almost the same as the ones needed (Table 3) to complete the derivations of the IRR(7) from the IRR(6) assuming that the first 3 columns of Table 3 are the same as the first 3 columns of Table 1. This last assumption can be verified from the computer program [4] output and is left as an assumption to be proved correct later. For example this assumption implies that state `3` and symbol `b` at the right must be followed by the new symbols `a`,b, or `c` on the right. In (8) this applies to rows 1 and 5. For row 1

LHS of IRR(6)		Symbol added	IRR used to complete the	
State	Symbol at opposite(R) end from pointer(L)	at pointer in the RHS of the IRR(6)	derivation of the IRR(7), which depend on other symbols	
3	b	a	$\mathtt{3\underline{a}} \rightarrow \mathtt{3b\_}$	
3	b	b	$\mathtt{3cc\underline{b}} \rightarrow \mathtt{3cbb\_}$	$\mathtt{3b\underline{b}} \rightarrow \mathtt{3cc\_}$
3	b	c	$\mathtt{3\underline{c}} \rightarrow \mathtt{3d\_}$	
4	a	b	$\mathtt{3cc\underline{b}} \rightarrow \mathtt{3cbb\_}$	$\mathtt{3b\underline{b}} \rightarrow \mathtt{3cc\_}$
4	a	d	$\mathtt{3cc\underline{d}} \rightarrow \mathtt{3cbd\_}$	$\mathtt{3b\underline{d}} \rightarrow \mathtt{4ca\_}$
4	c	b	$\mathtt{4\underline{b}} \rightarrow \mathtt{2c\_}$	$\mathtt{3d\underline{b}} \rightarrow \mathtt{3cb\_}$
4	b	d	$\mathtt{3cc\underline{d}} \rightarrow \mathtt{3cbd\_}$	$\mathtt{2\underline{d}} \rightarrow \mathtt{2\underline{d}}$
			$\mathtt{3b\underline{d}} \rightarrow \mathtt{4ca\_}$	

Table 3: Summary of the derivations of the IRR(7) from the IRR(6)

this gives the following results

$$\begin{aligned}
\mathtt{3\ldots baba} &\rightarrow \mathtt{3\ldots cc\underline{a}} \rightarrow \mathtt{3\ldots ccb_} \\
\mathtt{3\ldots babb} &\rightarrow \mathtt{3\ldots cc\underline{b}} \rightarrow \mathtt{3\ldots cbb_} \qquad (9)\\
\mathtt{3\ldots babc} &\rightarrow \mathtt{3\ldots cc\underline{c}} \rightarrow \mathtt{3\ldots ccd_}.
\end{aligned}$$

and the 3 rules to complete the derivations are respectively $\mathtt{3\underline{a}} \rightarrow \mathtt{3b\_}$, $\mathtt{3cc\underline{b}} \rightarrow \mathtt{3cbb\_}$, and $\mathtt{3\underline{c}} \rightarrow \mathtt{3d\_}$, which are identified easily. Doing this for all the outline derivations of rules in (8) gives results which can be summarised in Table 3. The complete list is given in (10). Column 4 of Tables 1-3 list all the possible final rules used to complete the derivations, and therefore indicate all the possible rightmost sequences of symbols in the RHS's of these derived rules.

The only difference between Table 3 and Table 1 is that in Table 3 the two rules of length 4 do not appear. It is interesting to note that in only these two cases is the RHS of the rule a subset of the RHS of another rule for the same values in columns 1-3. Therefore if the rule outlines indicated as obtained by the use of Table 3 are extended by one in length where possible according to Table 2, the result is the same as (8) and every rule in column 4 of Table 2 is used. When each of the types of rule indicated in (8) is extended by one again in length where possible according to Table 3, the result is the same as (10). In this list each of the rules in column 4 of Table 3 is used. This establishes the following theorem:

Theorem 2.1. *If Tables 2 and 3 are applied alternately to increase the length of the rules by 1 starting with the IRR(5) and using Table 2 to obtain the IRR(6), then sets of rules of arbitrary length for TM1 can be obtained.*

Furthermore this argument can be slightly extended to prove by induction that

Theorem 2.2. *Tables 2 and 3 summarise derivations of some of the rules of length* $\mathtt{n}+\mathtt{1}$ *from some of the rules of length* $\mathtt{n}$ *for* $\mathtt{n}$ *even and odd respectively for* $\mathtt{n} \geq \mathtt{4}$. *Column 4 of these Tables contains all the possible rules used to complete the derivations, and one application of one such rule is sufficient for each derivation.*

The proof for Table 3 follows from (8) by adding the symbol indicated in column 3 of Table 3 to each line of (8) that matches the state and symbol in columns 1 and 2 of Table 3, then noticing that one of the rules in column 4 of Table 3 allows the completion of the derivation in each case, and only one derivation step is required. All the rules in column 4 of Table 3 are used at least once. The proof for Table 2 is similar but requires the use of (10) in the place of (8).

$$
\begin{array}{lclcl}
\mathtt{3}\ldots\mathtt{baba} & \to & \mathtt{3}\ldots\mathtt{cc}\underline{\mathtt{a}} & \to & \mathtt{3}\ldots\mathtt{ccb\_} \\
\mathtt{3}\ldots\mathtt{babb} & \to & \mathtt{3}\ldots\mathtt{cc}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbb\_} \\
\mathtt{3}\ldots\mathtt{babc} & \to & \mathtt{3}\ldots\mathtt{cc}\underline{\mathtt{c}} & \to & \mathtt{3}\ldots\mathtt{ccd\_} \\
\mathtt{3}\ldots\mathtt{bcba} & \to & \mathtt{3}\ldots\mathtt{cb}\underline{\mathtt{a}} & \to & \mathtt{3}\ldots\mathtt{cbb\_} \\
\mathtt{3}\ldots\mathtt{bcbb} & \to & \mathtt{3}\ldots\mathtt{cb}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{ccc\_} \\
\mathtt{3}\ldots\mathtt{bcbc} & \to & \mathtt{3}\ldots\mathtt{cb}\underline{\mathtt{c}} & \to & \mathtt{3}\ldots\mathtt{cbd\_} \\
\mathtt{4}\ldots\mathtt{abab} & \to & \mathtt{3}\ldots\mathtt{ccb}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cccc\_} \\
\mathtt{4}\ldots\mathtt{abad} & \to & \mathtt{3}\ldots\mathtt{ccb}\underline{\mathtt{d}} & \to & \mathtt{4}\ldots\mathtt{ccca\_} \\
\mathtt{4}\ldots\mathtt{abbd} & \to & \mathtt{3}\ldots\mathtt{cbb}\underline{\mathtt{d}} & \to & \mathtt{4}\ldots\mathtt{cbca\_} \\
\mathtt{4}\ldots\mathtt{abcb} & \to & \mathtt{3}\ldots\mathtt{ccd}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cccb\_} \\
\mathtt{4}\ldots\mathtt{abab} & \to & \mathtt{3}\ldots\mathtt{cbbb}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbbcc\_} \\
\mathtt{4}\ldots\mathtt{abad} & \to & \mathtt{3}\ldots\mathtt{cbbb}\underline{\mathtt{d}} & \to & \mathtt{4}\ldots\mathtt{cbbca\_} \\
\mathtt{4}\ldots\mathtt{abbd} & \to & \mathtt{3}\ldots\mathtt{cbcc}\underline{\mathtt{d}} & \to & \mathtt{3}\ldots\mathtt{cbcbd\_} \\
\mathtt{4}\ldots\mathtt{abcb} & \to & \mathtt{3}\ldots\mathtt{cbbd}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbbcb\_} \\
\mathtt{4}\ldots\mathtt{cbab} & \to & \mathtt{3}\ldots\mathtt{cc}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbb\_} \\
\mathtt{4}\ldots\mathtt{cbad} & \to & \mathtt{3}\ldots\mathtt{cc}\underline{\mathtt{d}} & \to & \mathtt{3}\ldots\mathtt{cbd\_} \\
\mathtt{4}\ldots\mathtt{cbbd} & \to & \mathtt{2}\ldots\mathtt{dd}\underline{\mathtt{d}} & \to & \mathtt{2}\ldots\mathtt{dd}\underline{\mathtt{d}} \\
\mathtt{4}\ldots\mathtt{cbcb} & \to & \mathtt{4}\ldots\mathtt{ca}\underline{\mathtt{b}} & \to & \mathtt{2}\ldots\mathtt{cac\_} \\
\mathtt{4}\ldots\mathtt{cbab} & \to & \mathtt{3}\ldots\mathtt{cbb}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbcc\_} \\
\mathtt{4}\ldots\mathtt{cbad} & \to & \mathtt{3}\ldots\mathtt{cbb}\underline{\mathtt{d}} & \to & \mathtt{4}\ldots\mathtt{cbca\_} \\
\mathtt{4}\ldots\mathtt{cbbd} & \to & \mathtt{3}\ldots\mathtt{ccc}\underline{\mathtt{d}} & \to & \mathtt{3}\ldots\mathtt{ccbd\_} \\
\mathtt{4}\ldots\mathtt{cbcb} & \to & \mathtt{3}\ldots\mathtt{cbd}\underline{\mathtt{b}} & \to & \mathtt{3}\ldots\mathtt{cbcb\_}
\end{array}
\tag{10}
$$

2.2 General description of reverse computation rules and their usage

The basic problem with the above arguments is the lack of proofs that using Tables 2 and 3 as described, all the rules so generated are IRR for TM1 and that all the LHS's of the IRR are included. Because the IRR are infinite in number, a finite recursive description for them is desired if possible. Up to length 4 these LHS's have been found and are given in Section 2 in equations (2),(3) and (4). The computer programs [4] and [5] were used and are in agreement with these results that can be checked by hand, as will be shown by the end of this section.

The two requirements for IRR are that they are irreducible and that they are regular [3]. Irreducibility means that there are no redundant symbols, and is guaranteed for rules with the pointer starting at one end and ending at the other end of the string of symbols stated (i.e. rules of types LR or RL) and regularity means that the rule can be obtained from some rule of length 1 (a state-symbol pair and its RHS) by repeating a cycle in which (1) some new symbol is added in the RHS at the pointer at one end of the string of symbols (CS `x`) and (2) the computation goes as far as possible without adding a new symbol giving CS `y`. The new rule is $\mathtt{x} \rightarrow \mathtt{y}$. A CS is defined to be reachable if and only if it is the LHS or RHS of a regular rule as defined above. It is clear from this definition that for a CS `x` to be reachable, a reverse computation path must exist from `x` with the pointer passing every point in the string of symbols in `x`. Following the RHS in the above definition, such a CS is obtainable from a state-symbol pair by alternately adding symbols one at a time and going as far as possible with the computation. Reachability is satisfied for any CS that forms the LHS or the RHS of an IRR.

In this section a method for generating the IRR for TM1 based on these definitions will be followed. Reachability of a CS `x` with the pointer at one end can be checked by searching for backward TM steps, which when followed forwards satisfy the above definition. At each stage, none, one or more may be found, so generally a tree structure is formed. The computations going backwards along each possible branch can be terminated in one of 3 conditions: (1) the pointer reaches the opposite end of the string of symbols from its position in `x`, (2) the pointer reaches again the same end of the string of symbols as it was in `x`, or (3) the pointer is not at an end of the string of symbols and no preceding CS exists.

When condition (1) occurs, a new origin CS `y` for `x` has been obtained demonstrating that `x` is reachable. For, computing forward from this point `y` to `x`, it must have been the last time that the pointer was opposite where it is in `x`. From this a new derivation of reachability of `x` can be found by proceeding forward from `y`, adding the appropriate new symbol at the pointer whenever the pointer reaches a new position it has never reached before since CS $\mathtt{y}$.

If condition (2) occurs (CS $\mathtt{y}$) without condition (1) having yet occurred while searching backwards from $\mathtt{x}$, there is no point in continuing the derivation of that branch of the reverse rule. For suppose one branch of such a derivation is $\mathtt{x} \leftarrow \ldots \leftarrow \mathtt{y} \leftarrow \ldots$ where the CS $\mathtt{x}$ has the form $\mathtt{s}\underline{\alpha}\beta\gamma\ldots$ and $\mathtt{y}$ has the form $\mathtt{s}'\underline{\alpha'}\beta'\gamma'\ldots$ then running the computation forwards gives $\mathtt{y} \rightarrow \mathtt{x} \rightarrow \ldots$ where the computation continues beyond $\mathtt{x}$ because the pointer still has a symbol to be read at $\mathtt{x}$ so no new origin for $\mathtt{x}$ can be obtained from this argument (though one may exist as shown by another branch).

If in a branch, condition (3) occurs (CS $\mathtt{y}$) first while searching backwards, then because the pointer has not reached the opposite end of the string from where it is at $\mathtt{x}$, it cannot be used to prove that $\mathtt{x}$ is reachable because not all the string of symbols in $\mathtt{x}$ has been passed by the pointer in the computation $\mathtt{y} \rightarrow \mathtt{x}$, and no further searching backwards is possible.

If a CS $\mathtt{x}$ is reachable, this method will reveal the proof because there must be a forward computation path from a CS $\mathtt{y}$ of length 1 to $\mathtt{x}$ (adding symbols as needed) which has the pointer at every position in $\mathtt{x}$ at some point in the computation, so there is a point in this computation going backwards where the pointer first reaches the opposite point from where it was in $\mathtt{x}$ i.e. condition (1) has occurred. Because all branches have been followed in this search procedure, if no such proof of reachability can be obtained, then $\mathtt{x}$ is not reachable.

To avoid as much repetition as possible while carrying out many such derivations, small "equations" in the form of trees should be developed as above from short CS's and reused when possible to develop larger trees derived from longer CS's. For brevity for adding a single extra symbol to $\mathtt{x}$ at the opposite end from the pointer, it will be convenient to add a Greek symbol (e.g. $\alpha, \beta, \gamma, \delta$) to stand for any single symbol, in this example i.e. $\mathtt{a}$,$\mathtt{b}$,$\mathtt{c}$, or $\mathtt{d}$. Conditions on the Greek symbol can then be added on the relevant left facing arrows.

For any CS of length $\mathtt{n}$, let $\mathtt{S}(\mathtt{x})$ be the set of all the starting CS's which lead to it in condition (1) i.e. those that demonstrate the reachability of $\mathtt{x}$. The CS $\mathtt{x}$ is reachable if and only if $\mathtt{S}(\mathtt{x}) \neq \emptyset$. Then extend this as follows to length $\mathtt{n}+\mathtt{1}$ by adding each single symbol at the opposite end of the string of symbols from the pointer in $\mathtt{x}$ i.e. $\mathtt{x}\alpha$. The CS $\mathtt{x}\alpha$ is reachable if and only if there exists a member of $\mathtt{S}(\mathtt{x})\alpha$ that is reachable. The procedure is then to obtain from this the set of CS's that lead to $\mathtt{S}(\mathtt{x})\alpha$ i.e. $\mathtt{S}(\mathtt{x}\alpha)$. This of course depends on α and may or may not be empty. (The last step can usually be obtained using shorter reverse rules that are often needed repeatedly in the set of derivations as the examples will show.) To demonstrate this, in the proof that $\mathtt{x}\alpha$ is reachable, there must be a branch that ends (going backwards with the TM) with the pointer at the position of α so there is a first time (going backwards) that the pointer was next to α. The CS at this point is in $\mathtt{S(x)}\alpha$ for each such branch. Therefore the proof of reachability of $\mathtt{x}\alpha$ must involve reaching $\mathtt{S(x)}\alpha$ at some point in the derivation and the forward computation is in outline $\mathtt{S(x}\alpha) \to \mathtt{S(x)}\alpha \to \mathtt{x}\alpha$.

For a reverse rule of length $\mathtt{n}$ with the pointer at one end on the LHS, the branches ending in condition (2) will not lead to any irreducible rules of length $\mathtt{n}+\mathtt{1}$ by adding a symbol to each side etc. as above. In general however re-usable reverse rules do not necessarily start with the pointer at one end of the string, so the terms condition (1) and condition (2) do not apply, but if such a branch ends in condition (3), as part of a longer derivation, this is still the case and this branch cannot contribute to a proof of reachability. In order to be sure that, in a derivation of reachability of which a re-usable reverse rule forms a part, the computation will not continue beyond the point where either condition (1) or condition (2) first occurs, it is necessary that re-usable reverse rules will terminate when the pointer reaches either end of the string of symbols.

To illustrate all these concepts consider the reachability of $\mathtt{x} = \mathtt{3\underline{a}bab}$. This can be reached from only $\mathtt{4a\underline{c}ab}$ which can be reached from $\mathtt{2\underline{c}cab}$ (this branch now ends in condition 2) and $\mathtt{3ac\underline{b}b}$. $\mathtt{3ac\underline{b}b}$ can be reached from $\mathtt{2a\underline{a}bb}$ (this branch now ends in condition 3), $\mathtt{4a\underline{d}bb}$ and $\mathtt{4acb\underline{c}}$ (this branch now ends in condition 1). $\mathtt{4a\underline{d}bb}$ can be reached from $\mathtt{2\underline{c}dbb}$ (this branch now ends in condition 2) and $\mathtt{1ad\underline{b}b}$, which can be reached from only $\mathtt{4\underline{a}cbb}$ (condition 2) after 2 steps. This shows that $\mathtt{x}$ is reachable uniquely from $\mathtt{4acb\underline{c}}$ (after the branches that end in conditions 2 and 3 are deleted) as indicated in the third result of (26) for $\beta = \mathtt{a}$. Running the computation forwards we have $\mathtt{4\underline{c}} \to \mathtt{3\underline{b}b} \to \mathtt{4\underline{c}ab} \to \mathtt{3\underline{a}bab}$ after adding the extra symbols $\mathtt{b}$,$\mathtt{c}$ and $\mathtt{a}$ respectively at the pointer demonstrating the reachability of $\mathtt{3\underline{a}bab}$. This complete result can be expressed in the following "equation"

$$\mathtt{3\underline{a}bab} \leftarrow \begin{cases} \mathtt{2\underline{c}cab} \\ \mathtt{4acb\underline{c}} \\ \mathtt{2\underline{c}dbb} \\ \mathtt{4\underline{a}cbb} \end{cases} \tag{11}$$

in which only branches ending in condition 3 have been deleted. Note that the other three "origins" listed here do not count (condition 2) because their computations going forward have the pointer going beyond the right hand end. For example using the first result of (11) in condition 2, $\mathtt{2\underline{c}cab} \to \mathtt{3\underline{a}bab} \to \mathtt{3cccc\_}$, and this much computation would be done in one step in the definition of reachability, so this cannot be used to show that $\mathtt{3\underline{a}bab}$ is reachable. The same argument would have also worked if there had been some other CS with the pointer at the right hand end that was an origin for $\mathtt{2\underline{c}cab}$.

Now try using (11) and the general method to look for symbols α such that $\mathtt{3\underline{a}bab}\alpha$ is reachable. Clearly such an extra symbol added will not affect the status of any of these branches, so for example going back from $\mathtt{3\underline{a}baba}$ we come to $\mathtt{4acb\underline{c}a}$, which can be reached from $\mathtt{3acbc\underline{b}}$ in one step, so this is an origin demonstrating the reachability of $\mathtt{3\underline{a}baba}$. Thus $\mathtt{3acbc\underline{b}} \in \mathtt{S}(\mathtt{3\underline{a}baba})$. That this is the only element follows from the fact that the branches in condition (2) in (11) cannot lead to any more ways in which reachability can be arrived at after adding extra symbols on the right. For example $\mathtt{2\underline{c}cab} \to \mathtt{3\underline{a}bab}$ in (11) implies $\mathtt{2\underline{c}caba} \to \mathtt{3\underline{a}baba}$ and trying to go back from $\mathtt{2\underline{c}caba}$ without going "off the end" is impossible for the same reason that going back from $\mathtt{2\underline{c}cab}$ is i.e. $\mathtt{2\_c}$ has no preceding CS. Likewise $\mathtt{3\underline{a}babb}$ and $\mathtt{3\underline{a}babc}$ are both reachable from only $\mathtt{1acbc\underline{b}}$ and $\mathtt{3acbc\underline{d}}$ respectively. However after putting symbol $\mathtt{d}$ on the right, we have $\mathtt{4acb\underline{c}d} \to \mathtt{3\underline{a}babd}$ and $\mathtt{4acb\underline{c}d}$ is not reachable from a state-symbol pair because neither is $\mathtt{4b\_}$ (already obvious from (11)) or $\mathtt{4\_d}$, therefore $\mathtt{3\underline{a}babd}$ is not reachable.

Now TM1 will be systematically investigated to obtain the IRR($\mathtt{n}$) for all $\mathtt{n} > \mathtt{0}$ in a recursive description.

2.3 Systematic application of reverse computation rules to obtain the IRR for TM1

The first obvious step is to express the machine table in terms of reverse computation rules and summarised using place-holder Greek symbols as follows:

$$
\begin{array}{ll}
\mathtt{1\_}\alpha \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{2\underline{b}} & \mathtt{1}\alpha\mathtt{\_} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{4\underline{a}} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{1\underline{c}} \end{cases} \\[2ex]
\mathtt{2\_}\alpha \leftarrow \emptyset & \mathtt{2}\alpha\mathtt{\_} \begin{cases} \overset{\alpha=\mathtt{c}}{\longleftarrow} \begin{cases} \mathtt{1\underline{a}} \\ \mathtt{4\underline{b}} \end{cases} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{1\underline{d}} \end{cases} \\[2ex]
\mathtt{3\_}\alpha \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{4\underline{c}} & \mathtt{3}\alpha\mathtt{\_} \begin{cases} \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3\underline{a}} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \begin{cases} \mathtt{2\underline{a}} \\ \mathtt{4\underline{d}} \end{cases} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{3\underline{c}} \end{cases} \\[2ex]
\mathtt{4\_}\alpha \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3\underline{b}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{1\underline{b}} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{3\underline{d}} \end{cases} & \mathtt{4}\alpha\mathtt{\_} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{2\underline{c}}
\end{array}
\tag{12}
$$

2.3.1 Obtaining the LHS's of the IRR(2)

Extending this to longer rules first needs symbols to be added at the pointer. Starting with $\mathtt{1}\underline{\ }\alpha$ add an arbitrary symbol to get $\mathtt{1}\underline{\beta}\alpha$. Under what conditions is this both reachable and generating an irreducible rule? Reachability requires $\alpha = \mathtt{d}$. Irreducibility of the rule generated by it happens only if the first forward TM step from there is to the right, so this requires $\beta \in \{\mathtt{a}, \mathtt{c}, \mathtt{d}\}$. Likewise the CS $\mathtt{1}\alpha\underline{\beta}$ is reachable if and only if $\alpha \in \{\mathtt{a}, \mathtt{d}\}$ and leads to an irreducible rule if only if $\beta = \mathtt{b}$. This argument is easily extended to each LHS of (12) to give 7 sets of LHS's of IRR(2) which are listed together with their origins in (13). The origins are the final CS's in the search described above (trivial in these cases) that ends in condition (1):

$$
\begin{aligned}
&\mathtt{1}\underline{\beta}\mathtt{d} \leftarrow \mathtt{2}\beta\underline{\mathtt{b}} \text{ for } \beta \in \{\mathtt{a}, \mathtt{c}, \mathtt{d}\}\\
&\mathtt{1}\alpha\underline{\mathtt{b}} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{4}\underline{\mathtt{a}}\mathtt{b} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{1}\underline{\mathtt{c}}\mathtt{b} \end{cases}\\
&\mathtt{2}\alpha\underline{\mathtt{b}} \begin{cases} \overset{\alpha=\mathtt{c}}{\longleftarrow} \begin{cases} \mathtt{1}\underline{\mathtt{a}}\mathtt{b} \\ \mathtt{4}\underline{\mathtt{b}}\mathtt{b} \end{cases} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{1}\underline{\mathtt{d}}\mathtt{b} \end{cases}\\
&\mathtt{3}\underline{\beta}\mathtt{b} \leftarrow \mathtt{4}\beta\underline{\mathtt{c}} \text{ for } \beta \in \{\mathtt{a}, \mathtt{c}\}\\
&\mathtt{3}\alpha\underline{\beta} \begin{cases} \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3}\underline{\mathtt{a}}\beta \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \begin{cases} \mathtt{2}\underline{\mathtt{a}}\beta \\ \mathtt{4}\underline{\mathtt{d}}\beta \end{cases} \qquad \text{for } \beta \in \{\mathtt{b}, \mathtt{d}\} \\ \overset{\alpha=\mathtt{d}}{\longleftarrow} \mathtt{3}\underline{\mathtt{c}}\beta \end{cases}\\
&\mathtt{4}\underline{\beta}\alpha \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3}\beta\underline{\mathtt{b}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{1}\beta\underline{\mathtt{b}} \quad \text{for } \beta \in \{\mathtt{a}, \mathtt{b}, \mathtt{d}\} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{3}\beta\underline{\mathtt{d}} \end{cases}\\
&\mathtt{4}\mathtt{a}\underline{\mathtt{c}} \leftarrow \mathtt{2}\underline{\mathtt{c}}\mathtt{c}
\end{aligned} \tag{13}
$$

2.3.2 Obtaining the LHS's of the IRR(3)

The RHS's resulting from forward computation from each of the LHS CS's in (13) are obtained using TM1 giving the IRR in (2) of type LR or RL. Only these can generate irreducible rules for the next value of $\mathtt{n}$ by adding the an arbitrary symbol at the opposite end from the pointer in their LHS's and then checking for reachability of the resulting LHS's. The first set of CS's to be considered is $\mathtt{1}\underline{\beta}\mathtt{d}$ that generates a rule of type RL or LR if and only if $\mathtt{1}\underline{\beta}\mathtt{d} \rightarrow$ a CS with the pointer just beyond the right hand end. From (1) and (2) this happens only if $\beta = \mathtt{c}$. Tracing back the computation after adding the arbitrary symbol α to check for reachability gives the following:

$$\mathtt{1\underline{c}d}\alpha \leftarrow \mathtt{2c\underline{b}}\alpha \leftarrow \begin{cases} \mathtt{1\underline{a}b}\alpha \\ \mathtt{4\underline{b}b}\alpha \end{cases}. \tag{14}$$

Note that both the branches end in condition (2) and because the search is exhaustive, the LHS is not reachable. From (13) the next set is $\mathtt{1}\alpha\mathtt{\underline{b}}$ with $\alpha \in \{\mathtt{a},\mathtt{d}\}$. RL or LR type for the rule generated requires using (2) $\alpha = \mathtt{d}$ (the other case cycles) so the CS becomes $\mathtt{1d\underline{b}}$ and from (12)

$$\mathtt{1}\beta\mathtt{d\underline{b}} \leftarrow \mathtt{1}\beta\mathtt{\underline{c}b} \begin{cases} \overset{\beta=\mathtt{a}}{\leftarrow} \mathtt{4\underline{a}cb} \\ \overset{\beta=\mathtt{d}}{\leftarrow} \mathtt{1\underline{c}cb} \end{cases} \tag{15}$$

which shows that $\mathtt{1}\beta\mathtt{d\underline{b}}$ is reachable if and only if $\beta \in \{\mathtt{a},\mathtt{d}\}$. The next set to be considered in (13) is $\mathtt{2}\alpha\mathtt{\underline{b}}$ for $\alpha \in \{\mathtt{c},\mathtt{d}\}$ neither of which generate a rule of type RL or LR using (2). The CS's $\mathtt{3}\underline{\beta}\mathtt{b}$ with $\beta \in \{\mathtt{a},\mathtt{c}\}$, from (13), both generate rules of type RL or LR. Checking for reachability after adding the extra arbitrary symbol gives

$$\mathtt{3}\underline{\beta}\mathtt{b}\alpha \leftarrow \mathtt{4}\beta\mathtt{\underline{c}}\alpha \begin{cases} \overset{\beta=\mathtt{a}}{\leftarrow} \mathtt{2\underline{c}c}\alpha \\ \overset{\alpha=\mathtt{a}}{\leftarrow} \mathtt{3}\beta\mathtt{c\underline{b}} \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{1}\beta\mathtt{c\underline{b}} \\ \overset{\alpha=\mathtt{c}}{\leftarrow} \mathtt{3}\beta\mathtt{c\underline{d}} \end{cases} \tag{16}$$

so for each $\beta \in \{\mathtt{a},\mathtt{c}\}$ this gives 3 reachable CS's for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{c}\}$ i.e. 6 members of IRR(3). Of the CS's $\mathtt{3}\alpha\underline{\beta}$ with $\alpha \in \{\mathtt{b},\mathtt{c},\mathtt{d}\}$ and $\beta \in \{\mathtt{b},\mathtt{d}\}$, only $\mathtt{3c\underline{b}}$ and $\mathtt{3c\underline{d}}$ can generate rules of type RL or LR. Combining these and checking them for reachability gives, after adding one extra symbol α at the opposite end from the pointer gives,

$$\mathtt{3}\alpha\mathtt{c}\underline{\beta} \begin{cases} \leftarrow \mathtt{2}\alpha\mathtt{\underline{a}}\beta \begin{cases} \overset{\alpha=\mathtt{c}}{\leftarrow} \begin{cases} \mathtt{1\underline{a}a}\beta \\ \mathtt{4\underline{b}a}\beta \end{cases} \\ \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{1\underline{d}a}\beta \end{cases} \\ \leftarrow \mathtt{4}\alpha\mathtt{\underline{d}}\beta \begin{cases} \overset{\alpha=\mathtt{a}}{\leftarrow} \mathtt{2\underline{c}d}\beta \\ \overset{\beta=\mathtt{b}}{\leftarrow} \mathtt{1}\alpha\mathtt{d\underline{b}} \end{cases} \end{cases} \quad \text{for } \beta \in \{\mathtt{b},\mathtt{d}\} \tag{17}$$

where only the cases $\beta \in \{\mathtt{b},\mathtt{d}\}$ are to be considered. What has to be done now is to find every value of α such that there is a branch that ends in condition (1) thus establishing the reachability of $\mathtt{3}\alpha\mathtt{c}\underline{\beta}$. It is easy to see that this condition is independent of β. So for each $\alpha \in \{\mathtt{a},\mathtt{c},\mathtt{d}\}$, $\mathtt{3}\alpha\mathtt{c}\underline{\beta}$ is reachable for $\beta \in \{\mathtt{b},\mathtt{d}\}$ and these therefore all generate IRR. Repeating a similar argument starting from $\mathtt{4}\underline{\alpha}\beta$ for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\}$ and $\beta \in \{\mathtt{a},\mathtt{b},\mathtt{c}\}$ in (13) shows that only $\mathtt{4\underline{a}a}$, $\mathtt{4\underline{a}c}$ $\mathtt{4\underline{b}a}$, $\mathtt{4\underline{b}b}$, $\mathtt{4\underline{b}c}$, $\mathtt{4\underline{d}a}$, and $\mathtt{4\underline{d}c}$ generate rules of type RL or LR, and reachability of CS's of the form $4\underline{\alpha}\beta\gamma$ for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\}$, $\beta \in \{\mathtt{a},\mathtt{b},\mathtt{c}\}$, $\gamma \in \{\mathtt{a},\mathtt{b},\mathtt{c},\mathtt{d}\}$ is obtained from

$$4\underline{\alpha}\beta\gamma\begin{cases}\overset{\beta=\mathtt{a}}{\leftarrow} 3\alpha\underline{\mathtt{b}}\gamma\begin{cases}\overset{\alpha=\mathtt{b}}{\leftarrow} 3\underline{\mathtt{a}}\mathtt{b}\gamma\\ \overset{\alpha=\mathtt{d}}{\leftarrow} 3\underline{\mathtt{c}}\mathtt{b}\gamma\\ \overset{\gamma=\mathtt{b}}{\leftarrow} 4\alpha\mathtt{b}\underline{\mathtt{c}}\end{cases}\\ \overset{\beta=\mathtt{b}}{\leftarrow} 1\alpha\underline{\mathtt{b}}\gamma\begin{cases}\overset{\alpha=\mathtt{a}}{\leftarrow} 4\underline{\mathtt{a}}\mathtt{b}\gamma\\ \overset{\alpha=\mathtt{d}}{\leftarrow} 1\underline{\mathtt{c}}\mathtt{b}\gamma\\ \overset{\gamma=\mathtt{d}}{\leftarrow} 2\alpha\mathtt{b}\underline{\mathtt{b}}\end{cases}\\ \overset{\beta=\mathtt{c}}{\leftarrow} 3\alpha\underline{\mathtt{d}}\gamma\begin{cases}\overset{\alpha=\mathtt{b}}{\leftarrow} 3\underline{\mathtt{a}}\mathtt{d}\gamma\\ \overset{\alpha=\mathtt{d}}{\leftarrow} 3\underline{\mathtt{c}}\mathtt{d}\gamma\\ \overset{\gamma=\mathtt{b}}{\leftarrow} 4\alpha\mathtt{d}\underline{\mathtt{c}}\end{cases}\end{cases}\tag{18}$$

which follows from (12) using the general method described at the beginning of section 2.2. This shows that reachability only holds for the following LHS's (their origins are shown on the right): $4\underline{\alpha}\mathtt{ab} \leftarrow 4\alpha\mathtt{b}\underline{\mathtt{c}}$, $4\underline{\alpha}\mathtt{bd} \leftarrow 2\alpha\mathtt{b}\underline{\mathtt{b}}$ and $4\underline{\alpha}\mathtt{cb} \leftarrow 4\alpha\mathtt{d}\underline{\mathtt{c}}$ for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\}$. Combining these conditions shows that the LHS's of these IRR found are (with their origins on the right) $4\underline{\mathtt{a}}\mathtt{ab} \leftarrow 4\mathtt{ab}\underline{\mathtt{c}}$, $4\underline{\mathtt{a}}\mathtt{cb} \leftarrow 4\mathtt{ad}\underline{\mathtt{c}}$, $4\underline{\mathtt{b}}\mathtt{ab} \leftarrow 4\mathtt{bb}\underline{\mathtt{c}}$, $4\underline{\mathtt{b}}\mathtt{bd} \leftarrow 2\mathtt{bb}\underline{\mathtt{b}}$, $4\underline{\mathtt{b}}\mathtt{cb} \leftarrow 4\mathtt{bd}\underline{\mathtt{c}}$, $4\underline{\mathtt{d}}\mathtt{ab} \leftarrow 4\mathtt{db}\underline{\mathtt{c}}$, $4\underline{\mathtt{d}}\mathtt{cb} \leftarrow 4\mathtt{dd}\underline{\mathtt{c}}$ in agreement with (3). Finally $4\mathtt{a}\underline{\mathtt{c}}$ goes right showing that $4\alpha\mathtt{a}\underline{\mathtt{c}}$ cannot be the LHS of an IRR.

Collecting all these results for LHS's of the IRR(3) shows that they are in agreement with the results in (3) and can be expressed (including their origins i.e. only branches ending in condition (1)) by

$$\begin{array}{l}1\mathtt{ad}\underline{\mathtt{b}} \leftarrow 4\underline{\mathtt{a}}\mathtt{cb}\\ 1\mathtt{dd}\underline{\mathtt{b}} \leftarrow 1\underline{\mathtt{c}}\mathtt{cb}\\ \left.\begin{array}{l}3\underline{\beta}\mathtt{ba} \leftarrow 3\beta\mathtt{c}\underline{\mathtt{b}}\\ 3\underline{\beta}\mathtt{bb} \leftarrow 1\beta\mathtt{c}\underline{\mathtt{b}}\\ 3\underline{\beta}\mathtt{bc} \leftarrow 3\beta\mathtt{c}\underline{\mathtt{d}}\end{array}\right\} \text{ for } \beta \in \{\mathtt{a},\mathtt{c}\}\\ \left.\begin{array}{l}3\mathtt{ac}\underline{\beta} \leftarrow 2\underline{\mathtt{c}}\mathtt{d}\beta\\ 3\mathtt{cc}\underline{\beta} \leftarrow \begin{cases}1\underline{\mathtt{a}}\mathtt{a}\beta\\ 4\underline{\mathtt{b}}\mathtt{a}\beta\end{cases}\\ 3\mathtt{dc}\underline{\beta} \leftarrow 1\underline{\mathtt{d}}\mathtt{a}\beta\end{array}\right\} \text{ for } \beta \in \{\mathtt{b},\mathtt{d}\}\\ 4\underline{\alpha}\beta\mathtt{b} \leftarrow 4\alpha\left\{\begin{array}{l}\mathtt{b} \text{ if } \beta=\mathtt{a}\\ \mathtt{d} \text{ if } \beta=\mathtt{c}\end{array}\right\}\underline{\mathtt{c}} \text{ for } \alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\} \text{ and } \beta \in \{\mathtt{a},\mathtt{c}\}\\ 4\underline{\mathtt{b}}\mathtt{bd} \leftarrow 2\mathtt{bb}\underline{\mathtt{b}}\end{array}\quad . \tag{19}$$

2.3.3 Obtaining the LHS's of the IRR(4)

From the LHS's in (19), the corresponding RHS's in (3) of the IRR(3) can be obtained, noting which lead to IRR of type RL or LR. As before only these can be extended to LHS's of members of IRR(4) by the addition of a symbol at the opposite end of the string from the pointer in their LHS's.

In (3) $\mathtt{1ad\underline{b}} \to \mathtt{3ccb\_}$ in IRR(3) has type RR therefore the rule derived from the LHS $\mathtt{1}\alpha\mathtt{ad\underline{b}}$ is reducible, but $\mathtt{1dd\underline{b}}$ goes left in (3) so it could generate an IRR. From (19) and (12),

$$\mathtt{1}\alpha\mathtt{dd\underline{b}} \leftarrow \mathtt{1}\alpha\mathtt{\underline{c}cb} \begin{cases} \overset{\alpha=\mathtt{a}}{\leftarrow} \mathtt{4\underline{a}ccb} \\ \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{1\underline{c}ccb} \end{cases} \tag{20}$$

so both these are reachable and so are LHS's of IRR(4).

The next set $\mathtt{3}\underline{\beta}\mathtt{b}\alpha$ all go right in (3) and from (19),

$$\mathtt{3}\underline{\beta}\mathtt{b}\alpha\gamma \begin{cases} \overset{\alpha=\mathtt{a}}{\leftarrow} \mathtt{3}\beta\mathtt{c\underline{b}}\gamma \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{1}\beta\mathtt{c\underline{b}}\gamma \\ \overset{\alpha=\mathtt{c}}{\leftarrow} \mathtt{3}\beta\mathtt{c\underline{d}}\gamma \end{cases} \quad \text{for } \beta \in \{\mathtt{a},\mathtt{c}\}. \tag{21}$$

The search for origins continues from the RHS's in (21) in the next set of equations that are obtained by repeatedly using (12):

$$\mathtt{3}\beta\mathtt{c\underline{b}}\gamma \left\{ \begin{array}{l} \leftarrow \mathtt{2}\beta\mathtt{\underline{a}b}\gamma \left\{ \begin{array}{l} \overset{\beta=\mathtt{c}}{\leftarrow} \left\{ \begin{array}{l} \mathtt{1\underline{a}ab}\gamma \\ \mathtt{4\underline{b}ab}\gamma \end{array} \right. \\ \overset{\beta=\mathtt{d}}{\leftarrow} \mathtt{1\underline{d}ab}\gamma \end{array} \right. \\ \leftarrow \mathtt{4}\beta\mathtt{\underline{d}b}\gamma \left\{ \begin{array}{l} \overset{\beta=\mathtt{a}}{\leftarrow} \mathtt{2\underline{c}db}\gamma \\ \leftarrow \mathtt{1}\beta\mathtt{d\underline{b}}\gamma \left\{ \begin{array}{l} \overset{\gamma=\mathtt{d}}{\leftarrow} \mathtt{2}\beta\mathtt{db\underline{b}} \\ \leftarrow \mathtt{1}\beta\mathtt{\underline{c}b}\gamma \left\{ \begin{array}{l} \overset{\beta=\mathtt{a}}{\leftarrow} \mathtt{4\underline{a}cb}\gamma \\ \overset{\beta=\mathtt{d}}{\leftarrow} \mathtt{1\underline{c}cb}\gamma \end{array} \right. \end{array} \right. \end{array} \right. \\ \overset{\gamma=\mathtt{b}}{\leftarrow} \mathtt{4}\beta\mathtt{cb\underline{c}} \end{array} \right. \tag{22}$$

$$\mathtt{1}\beta\mathtt{c\underline{b}}\gamma \overset{\gamma=\mathtt{d}}{\leftarrow} \mathtt{2}\beta\mathtt{cb\underline{b}}$$

$$\mathtt{3}\beta\mathtt{c\underline{d}}\gamma \left\{ \begin{array}{l} \leftarrow \mathtt{2}\beta\mathtt{\underline{a}d}\gamma \left\{ \begin{array}{l} \overset{\beta=\mathtt{c}}{\leftarrow} \left\{ \begin{array}{l} \mathtt{1\underline{a}ad}\gamma \\ \mathtt{4\underline{b}ad}\gamma \end{array} \right. \\ \overset{\beta=\mathtt{d}}{\leftarrow} \mathtt{1\underline{d}ad}\gamma \end{array} \right. \\ \leftarrow \mathtt{4}\beta\mathtt{\underline{d}d}\gamma \overset{\beta=\mathtt{a}}{\leftarrow} \mathtt{2\underline{c}dd}\gamma \\ \overset{\gamma=\mathtt{b}}{\leftarrow} \mathtt{4}\beta\mathtt{cd\underline{c}} \end{array} \right.$$

Therefore a CS of the form $\mathtt{3}\underline{\beta}\mathtt{b}\alpha\gamma$ is reachable if and only if $\alpha = \mathtt{a}$ and $\gamma \in \{\mathtt{b},\mathtt{d}\}$ or $\alpha = \mathtt{b}$ and $\gamma = \mathtt{d}$ or $\alpha = \mathtt{c}$ and $\gamma = \mathtt{b}$ i.e. only the following sets of CS's of the form $\mathtt{3}\underline{\beta}\mathtt{b}\alpha\gamma$ are reachable: $\mathtt{3}\underline{\beta}\mathtt{ba}\left\{ \begin{array}{c} \mathtt{b} \\ \mathtt{d} \end{array} \right\}$, $\mathtt{3}\underline{\beta}\mathtt{bbd}$, and $\mathtt{3}\underline{\beta}\mathtt{bcb}$, and the condition $\beta \in \{\mathtt{a},\mathtt{c}\}$ comes from (19). Extending the $\mathtt{3}\gamma\mathtt{c}\underline{\beta}$ for $\gamma \in \{\mathtt{a},\mathtt{c},\mathtt{d}\}$ and $\beta \in \{\mathtt{b},\mathtt{d}\}$ to be LHS's of IRR(4) members, note that of these only the following generate IRR(3) of type RL in (3): $\mathtt{3dc\underline{b}}$ and $\mathtt{3dc\underline{d}}$. With the extra arbitrary symbol, tracing these back from the result in (19) to find which are reachable gives

$$\mathtt{3}\delta\mathtt{dc}\underline{\beta} \leftarrow \mathtt{1}\delta\mathtt{\underline{d}a}\beta \begin{cases} \overset{\delta=\mathtt{a}}{\longleftarrow} \mathtt{4\underline{a}da}\beta \\ \overset{\delta=\mathtt{d}}{\longleftarrow} \mathtt{1\underline{c}da}\beta \end{cases} \quad \text{for } \beta \in \{\mathtt{b},\mathtt{d}\} \tag{23}$$

which shows that $\mathtt{3}\delta\mathtt{dc}\underline{\beta}$ for $\beta \in \{\mathtt{b},\mathtt{d}\}$ and $\delta \in \{\mathtt{a},\mathtt{d}\}$ are LHS's of IRR(4). All the CS's $\mathtt{4}\underline{\alpha}\beta\mathtt{b}$ for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\}$ and $\beta \in \{\mathtt{a},\mathtt{c}\}$ go right in (3) and using (19)

$$\mathtt{4}\underline{\alpha}\beta\mathtt{b}\gamma \leftarrow \mathtt{4}\alpha\mathtt{x\underline{c}}\gamma \begin{cases} \overset{\gamma=\mathtt{a}}{\longleftarrow} \mathtt{3}\alpha\mathtt{xc\underline{b}} \\ \overset{\gamma=\mathtt{b}}{\longleftarrow} \mathtt{1}\alpha\mathtt{xc\underline{b}} \\ \overset{\gamma=\mathtt{c}}{\longleftarrow} \mathtt{3}\alpha\mathtt{xc\underline{d}} \end{cases} \text{for } \beta \in \{\mathtt{a},\mathtt{c}\} \text{ and } \alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\} \tag{24}$$

where

$$\mathtt{x} = \left\{ \begin{array}{l} \mathtt{b} \text{ if } \beta = \mathtt{a} \\ \mathtt{d} \text{ if } \beta = \mathtt{c} \end{array} \right\}. \tag{25}$$

Therefore $\mathtt{4}\underline{\alpha}\beta\mathtt{b}\gamma$ are LHS's of IRR(4) for $\alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\}$ and $\beta \in \{\mathtt{a},\mathtt{c}\}$ and $\gamma \in \{\mathtt{a},\mathtt{b},\mathtt{c}\}$. Finally in (3) $\mathtt{4\underline{b}bd}$ goes to the halting CS $\mathtt{2dd\underline{d}}$ so cannot generate any IRR(4). This explains all the LHS's of IRR(4). These are all listed here with their origins of which there is now only one in each case:

$$\begin{array}{l}
\mathtt{1add\underline{b}} \leftarrow \mathtt{4\underline{a}ccb} \\
\mathtt{1ddd\underline{b}} \leftarrow \mathtt{1\underline{c}ccb} \\
\left.\begin{array}{l}
\mathtt{3}\underline{\beta}\mathtt{ba}\gamma \begin{cases} \overset{\gamma=\mathtt{b}}{\longleftarrow} \mathtt{4}\beta\mathtt{cb\underline{c}} \\ \overset{\gamma=\mathtt{d}}{\longleftarrow} \mathtt{2}\beta\mathtt{db\underline{b}} \end{cases} \\
\mathtt{3}\underline{\beta}\mathtt{bbd} \leftarrow \mathtt{2}\beta\mathtt{cb\underline{b}} \\
\mathtt{3}\underline{\beta}\mathtt{bcb} \leftarrow \mathtt{4}\beta\mathtt{cd\underline{c}}
\end{array}\right\} \text{for } \beta \in \{\mathtt{a},\mathtt{c}\} \\
\left.\begin{array}{l}
\mathtt{3adc}\underline{\beta} \leftarrow \mathtt{4\underline{a}da}\beta \\
\mathtt{3ddc}\underline{\beta} \leftarrow \mathtt{1\underline{c}da}\beta
\end{array}\right\} \text{for } \beta \in \{\mathtt{b},\mathtt{d}\} \\
\left.\begin{array}{l}
\mathtt{4}\underline{\alpha}\beta\mathtt{ba} \leftarrow \mathtt{3}\alpha\mathtt{xc\underline{b}} \\
\mathtt{4}\underline{\alpha}\beta\mathtt{bb} \leftarrow \mathtt{1}\alpha\mathtt{xc\underline{b}} \\
\mathtt{4}\underline{\alpha}\beta\mathtt{bc} \leftarrow \mathtt{3}\alpha\mathtt{xc\underline{d}}
\end{array}\right\} \begin{array}{l} \text{for } \alpha \in \{\mathtt{a},\mathtt{b},\mathtt{d}\} \text{ and } \beta \in \{\mathtt{a},\mathtt{c}\} \\ \text{and } \mathtt{x} \text{ is given by (25)} \end{array}
\end{array} \tag{26}$$

2.3.4 Obtaining the LHS's of the IRR(n) for n > 4

The results in this section are not written in the order that they were found because during the calculations it was found that short re-usable reverse rules could be used to efficiently complete all the proofs in a manner similar to the methods of the preceding sections. Therefore in this list of reverse rules, it is not obvious which symbols were to be added in the several stages of elongation. They were chosen only when necessary to complete the proofs of the main results for the IRR for length $> \mathtt{4}$ at the end of this section.

The first step is to note that from (4), the only LHS's in (26) leading to IRR of type LR or RL are $\mathtt{3}\underline{\beta}\mathtt{ba}\gamma$, $\mathtt{3}\underline{\beta}\mathtt{bbd}$, $\mathtt{3}\underline{\beta}\mathtt{bcb}$, for all the values of β and γ, and all the LHS's starting in state $\underline{4}$.

Starting from the first of these LHS's, the reachability of the CS $\mathtt{3}\underline{\mathtt{a}}\mathtt{bab}\alpha$ must be derived from the reachability of $\mathtt{4acb}\underline{\mathtt{c}}\alpha$. From (12) using the reachability of $\mathtt{4\_}\alpha$ it follows that

$$\mathtt{4b}\underline{\mathtt{c}}\alpha \begin{cases} \stackrel{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3bc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{1bc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{3bc}\underline{\mathtt{d}} \end{cases} . \tag{27}$$

where the $\mathtt{b}$ on the left implies that a reverse TM step to the left is not possible. From (26) and (27), the result for the origins for $\mathtt{3}\underline{\mathtt{a}}\mathtt{bab}\alpha$ is given by

$$\mathtt{3}\underline{\mathtt{a}}\mathtt{bab}\alpha \leftarrow \mathtt{4acb}\underline{\mathtt{c}}\alpha \begin{cases} \stackrel{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3acbc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{1acbc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{3acbc}\underline{\mathtt{d}} \end{cases} . \tag{28}$$

Equation (27) is the first of the re-usable reverse rules to be derived. Proceeding in the same way, the next available result of (26) for $\beta = \mathtt{a}$ is $\mathtt{3}\underline{\mathtt{a}}\mathtt{bad} \leftarrow \mathtt{2adb}\underline{\mathtt{b}}$, so for each symbol α, $\mathtt{3}\underline{\mathtt{a}}\mathtt{bad}\alpha \leftarrow \mathtt{2adb}\underline{\mathtt{b}}\alpha$ and $\mathtt{2adb}\underline{\mathtt{b}}\alpha$ is not reachable because from (12), $\mathtt{2b\_}$ and $\mathtt{2\_}\alpha$ for each $\alpha \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$ are not reachable from any state-symbol pair, i.e.

$$\mathtt{2b}\underline{\mathtt{b}}\alpha \leftarrow \emptyset, \tag{29}$$

and $\mathtt{3}\underline{\mathtt{a}}\mathtt{bad}\alpha$ is not reachable for every α. The following reverse rule was likewise obtained using the other available results in (26) as the starting points, during the derivations of all the LHS's of the IRR(5):

$$\mathtt{4d}\underline{\mathtt{c}}\alpha \begin{cases} \stackrel{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3dc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{1dc}\underline{\mathtt{b}} \\ \stackrel{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{3dc}\underline{\mathtt{d}} \end{cases} . \tag{30}$$

Note that this derivation requires checking from (12) that $\mathtt{4d\_}$ is not reachable. Similarly for the following

$$\mathtt{3c}\underline{\mathtt{b}}\alpha \begin{cases} \longleftarrow \quad \mathtt{2}\underline{\mathtt{a}}\mathtt{b}\alpha \\ \stackrel{\alpha=\mathtt{b}}{\longleftarrow} \quad \mathtt{4cb}\underline{\mathtt{c}} \\ \longleftarrow \quad \mathtt{4}\underline{\mathtt{d}}\mathtt{b}\alpha \leftarrow \mathtt{1d}\underline{\mathtt{b}}\alpha \end{cases} \tag{31}$$

from which follows again using (12) repeatedly

$$\mathtt{3bc\underline{b}}\alpha \begin{cases} \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4bcb\underline{c}} \\ \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{2bdb\underline{b}} \end{cases} \tag{32}$$

and similarly

$$\mathtt{3dc\underline{b}}\alpha \begin{cases} \leftarrow \mathtt{1\underline{d}ab}\alpha \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4dcb\underline{c}} \\ \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{2ddb\underline{b}} \\ \leftarrow \mathtt{1\underline{c}cb}\alpha \end{cases} . \tag{33}$$

From (12)

$$\mathtt{1\underline{b}}\alpha \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{2b\underline{b}} \tag{34}$$

which implies

$$\mathtt{1c\underline{b}}\alpha \overset{\alpha=\mathtt{d}}{\leftarrow} \mathtt{2cb\underline{b}}. \tag{35}$$

Again

$$\mathtt{3c\underline{d}}\alpha \begin{cases} \leftarrow \mathtt{2\underline{a}d}\alpha \\ \leftarrow \mathtt{4\underline{d}d}\alpha \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4cd\underline{c}} \end{cases} \tag{36}$$

which implies

$$\mathtt{3bc\underline{d}}\alpha \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4bcd\underline{c}} \tag{37}$$

and

$$\mathtt{3dc\underline{d}}\alpha \begin{cases} \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4dcd\underline{c}} \\ \leftarrow \mathtt{2d\underline{a}d}\alpha \leftarrow \mathtt{1\underline{d}ad}\alpha \end{cases} \tag{38}$$

and

$$\mathtt{3cdc\underline{d}}\alpha \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{4cdcd\underline{c}} \tag{39}$$

because $\mathtt{1c\_}$ and $\mathtt{1\_a}$ are both unreachable and the lower branch of (38) now ends in condition (3).

A summary of the derivations of the LHS's of the IRR(5) from those of the IRR(4) is contained in Table 4. Each row of the table represents one or more separate derivations starting with a CS having the specified state and symbols at its right hand end. The number of symbols indicated is not necessarily the minimum needed to show which reverse rule applies. Extra symbols were added by reference to the computer-generated list of IRR so that the general induction argument to be described below works, thus the first two columns of Table 4 are part of a theorem to be proved by induction on $\mathtt{n}$. The last column has all the symbols that can be added on the right, i.e. the α above. For example the first row of Table 4 starts with any CS of the form $\mathtt{3\ldots ab}$ (each of these has an origin of the form $\mathtt{4\ldots b\underline{c}}$), as the LHS of an IRR of even length and shows how it can be extended in length by 1 by adding a symbol $\alpha \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ to generate reachable CS's using the reverse rule (27):

$$\mathtt{3\ldots ab}\alpha \leftarrow \mathtt{4\ldots b\underline{c}}\alpha \begin{cases} \stackrel{\alpha=\mathtt{a}}{\leftarrow} & \mathtt{3\ldots bc\underline{b}} \\ \stackrel{\alpha=\mathtt{b}}{\leftarrow} & \mathtt{1\ldots bc\underline{b}} \\ \stackrel{\alpha=\mathtt{c}}{\leftarrow} & \mathtt{3\ldots bc\underline{d}} \end{cases} . \tag{40}$$

Then the result for $\alpha = \mathtt{a}, \mathtt{b}, \mathtt{c}$ are consistent with the first three rows of Table 5 respectively.

LHS of IRR($\mathtt{n}$)	Origin	Reverse rule	Set of Symbols	Condition
$\mathtt{3\ldots ab}$	$\mathtt{4\ldots b\underline{c}}$	(27)	$\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$	
$\mathtt{3\ldots ad}$	$\mathtt{2\ldots b\underline{b}}$	(29)	$\emptyset$	
$\mathtt{3\ldots bd}$	$\mathtt{2\ldots b\underline{b}}$	(29)	$\emptyset$	
$\mathtt{3\ldots bcb}$	$\mathtt{4\ldots cd\underline{c}}$	(30)	$\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$	
$\mathtt{4\ldots aba}$	$\mathtt{3\ldots bc\underline{b}}$	(32)	$\{\mathtt{b}, \mathtt{d}\}$	
$\mathtt{4\ldots abc}$	$\mathtt{3\ldots bc\underline{d}}$	(37)	$\{\mathtt{b}\}$	
$\mathtt{4\ldots bb}$	$\mathtt{1\ldots c\underline{b}}$	(35)	$\{\mathtt{d}\}$	
$\mathtt{4\ldots cbc}$	$\mathtt{3\ldots dc\underline{d}}$	(38)	$\{\mathtt{b}\}$	
$\mathtt{4\ldots cba}$	$\mathtt{3\ldots dc\underline{b}}$	(33)	$\{\mathtt{b}, \mathtt{d}\}$	

Table 4: Summary of the derivations of the LHS's of the IRR($\mathtt{n}+\mathtt{1}$) from the LHS's of the IRR($\mathtt{n}$) for $\mathtt{n} \geq \mathtt{4}$ and $\mathtt{n}$ even

The corresponding table for $\mathtt{n}$ odd is as follows:

LHS of IRR($\mathtt{n}$)	Origin	Reverse rule	Set of Symbols
$\mathtt{3\ldots aba}$	$\mathtt{3\ldots bc\underline{b}}$	(32)	$\{\mathtt{b}, \mathtt{d}\}$
$\mathtt{3\ldots bb}$	$\mathtt{1\ldots c\underline{b}}$	(35)	$\{\mathtt{d}\}$
$\mathtt{3\ldots abc}$	$\mathtt{3\ldots bc\underline{d}}$	(37)	$\{\mathtt{b}\}$
$\mathtt{3\ldots cba}$	$\mathtt{3\ldots dc\underline{b}}$	(33)	$\{\mathtt{b}, \mathtt{d}\}$
$\mathtt{3\ldots cbc}$	$\mathtt{3\ldots dc\underline{d}}$	(38)	$\{\mathtt{b}\}$
$\mathtt{4\ldots ab}$	$\mathtt{4\ldots b\underline{c}}$	(27)	$\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$
$\mathtt{4\ldots ad}$	$\mathtt{2\ldots b\underline{b}}$	(29)	$\emptyset$
$\mathtt{4\ldots bcb}$	$\mathtt{4\ldots cd\underline{c}}$	(30)	$\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$
$\mathtt{4\ldots bd}$	$\mathtt{2\ldots b\underline{b}}$	(29)	$\emptyset$

.

Table 5: Summary of the derivation of the IRR($\mathtt{n}+\mathtt{1}$) from the LHS's of the IRR($\mathtt{n}$) for $\mathtt{n} \geq \mathtt{5}$ and $\mathtt{n}$ odd

Let $\mathtt{T3}(\mathtt{n})$ be an abbreviation for Table 3 applied to rules of length $\mathtt{n}$ and mean the following:

> For each line of Table 3 defined by state S, symbols α and β and the set of rules R respectively, if a rule $\mathtt{r_1}$ of type LR and length n has state S in its LHS, symbol at the right hand end α, then it can be used to derive a rule $\mathtt{r_2}$ of type LR and length $\mathtt{n}+\mathtt{1}$ by adding the symbol β to the left and right of $\mathtt{r_1}$ on its right hand end and then applying in a single step one of the rules of R to this RHS.

The meaning of T2(n) is likewise and refers to Table 2. The meaning of T4(n) is as follows:

> For each line of Table 4 the first two columns are the CS outlines and will be denoted by c1, c2 respectively. If c1 matches the LHS of an IRR r of type LR and length n then c2 matches the unique origin of r with the pointer at the right (condition 1) according to the method above for searching for origins. There can be other origins with the pointer at the left but these end in condition 2 and do not lead to reachability arguments because new symbols are being added only at the right.

The meaning of T5(n) is likewise and refers to Table 5.

The importance of this is that, using the remaining elements in the lines of Tables 4 and 5 i.e. the reverse rule rr and set of symbols s respectively, according to the reverse rule rr and the origin c2, s is the set of all symbols any one of which can be applied to the LHS of r on its right to obtain the LHS of r1, a rule of length $\mathtt{n}+\mathtt{1}$ that must be regular i.e. its LHS is reachable because an origin for it has been found.

Theorem 2.3. *If* T3(n) *and* T4(n) *then the rule outlines and added symbols of Table 4 determine the set of LHS's of all the IRR*(n + 1) *from the set of LHS's of all the IRR*(n)*. This is done by extending on the right by any one symbol in column 4 of Table 4, the LHS's of any member of IRR*(n) *that match the LHS outline in column 1 of the same row of Table 4.*

The same is true with Table 3 (T3) replaced by Table 2 (T2) and Table 4 (T4) replaced by Table 5 (T5).

Proof. For example if 3 . . . ab matches the LHS of an IRR r of length n and type LR then by T4(n), $4\ldots\mathtt{b}\underline{\mathtt{c}}$ matches the unique origin in condition 1 of r and by adding symbols on the right of r and using (27), c1 = 3 . . . aba, c2 = 3 . . . abb and c3 = 3 . . . abc (and only these extensions of 3 . . . ab) match reachable LHS's of rules of length $\mathtt{n}+\mathtt{1}$, because an origin was found for each. Also by T3(n), (actually the first 3 rows of Table 3) the CS outlines c1,c2 and c3 each lead to a final CS giving a rule which is irreducible (of type LR). Thus c1, c2 and c3 generate IRR of length $\mathtt{n}+\mathtt{1}$ and are the only IRR obtainable by extending r by a single symbol.

To verify this for all cases it is simply necessary to note that every triple consisting of (1) the state in column 1 of Table 4, (2) the rightmost symbol in column 1 of Table 4 and (3) any of the added symbols in column 4 of Table 4 appear in columns 1, 2 and 3 of Table 3 respectively, and likewise with Table 3 replaced by Table 2, and Table 4 replaced by Table 5. $\square$

Theorem 2.4.

$$[\mathtt{T3}(\mathtt{n}) \textit{ and } \mathtt{T4}(\mathtt{n})] \Rightarrow \mathtt{T5}(\mathtt{n}+\mathtt{1}) \tag{41}$$

Proof. Roughly stated, each line of Table 5 must be consistent with every line of Table 4 with the addition of each applicable symbol and use of the appropriate reverse rule. In this proof I will describe in detail the argument for proving that one that line of $\mathtt{T5}(\mathtt{n}+\mathtt{1})$ follows from $\mathtt{T4}(\mathtt{n})$. To complete the proof, similar arguments must be made for all the other lines of $\mathtt{T5}(\mathtt{n}+\mathtt{1})$.

First note that the conclusion of Theorem 2.3 applies. Consider showing $\mathtt{3}\ldots\mathtt{bb} \leftarrow \mathtt{1}\ldots\mathtt{c}\underline{\mathtt{b}}$, the second line of $\mathtt{T5}$ for rules of length $\mathtt{n}+\mathtt{1}$. The LHS of this arises from adding the symbol $\mathtt{b}$ to $\mathtt{3}\ldots\mathtt{b}$ of length $\mathtt{n}$, and this matches the following LHS's in Table 4: $\mathtt{3}\ldots\mathtt{ab}$, $\mathtt{3}\ldots\mathtt{bcb}$, so using Theorem 2.3, any CS in $\mathtt{3}\ldots\mathtt{bb}$ of length $\mathtt{n}+\mathtt{1}$ that is the LHS of an IRR must arise from adding $\mathtt{b}$ to a CS in either $\mathtt{3}\ldots\mathtt{ab}$ or $\mathtt{3}\ldots\mathtt{bcb}$ of length $\mathtt{n}$. For both cases (for this argument to work, at least one of these two statements must be true), the symbol $\mathtt{b}$ can be added to generate a reachable CS according to the reverse rule used to derive the origins. Actually we have $\mathtt{3}\ldots\mathtt{abb} \leftarrow \mathtt{4}\ldots\mathtt{b}\underline{\mathtt{c}}\mathtt{b} \leftarrow \mathtt{1}\ldots\mathtt{bc}\underline{\mathtt{b}}$ by (27) and $\mathtt{3}\ldots\mathtt{bcbb} \leftarrow \mathtt{4}\ldots\mathtt{cd}\underline{\mathtt{c}}\mathtt{b} \leftarrow \mathtt{1}\ldots\mathtt{cdc}\underline{\mathtt{b}}$ by (30) and in each case the origin matches $\mathtt{1}\ldots\mathtt{c}\underline{\mathtt{b}}$. Note that all preceding CS's are always given on the RHS's of the left arrows and only a single origin in condition (1) is possible in each case. Therefore the pattern $\mathtt{1}\ldots\mathtt{c}\underline{\mathtt{b}}$ matches all possible origins in condition (1) of LHS's of IRR matching $\mathtt{3}\ldots\mathtt{bb}$ of length $\mathtt{n}+\mathtt{1}$. $\square$

Theorem 2.5.

$$[\mathtt{T2}(\mathtt{n}) \textit{ and } \mathtt{T5}(\mathtt{n})] \Rightarrow \mathtt{T4}(\mathtt{n}+\mathtt{1}) \tag{42}$$

Proof. The proof is exactly as above with $\mathtt{T4}$ and $\mathtt{T5}$ exchanged and $\mathtt{T3}$ and $\mathtt{T2}$ exchanged. $\square$

The main results in subsection 2.1 can be written as

$$\mathtt{T3}(4) \text{ and } \mathtt{T2}(5), \tag{43}$$

$$[\mathtt{T3}(\mathtt{n}) \text{ and } \mathtt{T2}(\mathtt{n}+\mathtt{1})] \Rightarrow \mathtt{T3}(\mathtt{n}+\mathtt{2}) \text{ for } \mathtt{n} \geq \mathtt{4}, \tag{44}$$

and

$$[\mathtt{T3}(\mathtt{n}) \text{ and } \mathtt{T2}(\mathtt{n}+\mathtt{1}) \text{ and } \mathtt{T3}(\mathtt{n}+\mathtt{2})] \Rightarrow \mathtt{T2}(\mathtt{n}+\mathtt{3}) \text{ for } \mathtt{n} \geq \mathtt{4}. \tag{45}$$

From (43), (44) and (45) it is easy to show that both

$$\texttt{T3}(\texttt{n}) \text{ for } \texttt{n} \text{ even and } \geq \texttt{4}, \text{ and} \tag{46}$$

$$\texttt{T2}(\texttt{n}) \text{ for } \texttt{n} \text{ odd and } \geq \texttt{5}. \tag{47}$$

The result

$$\texttt{T4}(\texttt{4}) \tag{48}$$

is Table 4 for $\texttt{n} = \texttt{4}$ and is easily seen to follow from (26). Putting $\texttt{n}$ even and $\geq \texttt{4}$ into (41) gives

$$\texttt{T4}(\texttt{n}) \Rightarrow \texttt{T5}(\texttt{n}+\texttt{1}) \text{ for } \texttt{n} \text{ even and } \geq \texttt{4}, \tag{49}$$

and putting $\texttt{n}$ odd and $\geq \texttt{5}$ into (42) gives

$$\texttt{T5}(\texttt{n}) \Rightarrow \texttt{T4}(\texttt{n}+\texttt{1}) \text{ for } \texttt{n} \text{ odd and } \geq \texttt{5}. \tag{50}$$

Finally from (48) and (49) and (50) it is easy to show that

$$\texttt{T4}(\texttt{n}) \text{ for } \texttt{n} \text{ even and } \geq \texttt{4} \tag{51}$$

and

$$\texttt{T5}(\texttt{n}) \text{ for } \texttt{n} \text{ odd and } \geq \texttt{5}. \tag{52}$$

The final results of this section are (46), (47), (51), and (52).

2.4 Global analysis of the TM

After some IRR have been applied to an initial tape configuration of any TM, let $\texttt{k}$ be the length of the next IRR to be applied to continue the computation and let $\texttt{n}$ be the length of the tape read by the TM including the last symbol read to determine the next IRR to be applied. Then in general $\texttt{n} \geq \texttt{k}$.

In the following argument for simplicity I shall ignore the possibility of the TM halting or entering into a stationary cycle. At any time these events could happen if an appropriate IRR rule (2 halting and and 2 stationary cycles for $\texttt{k} = \texttt{2}$, and one halting for each of $\texttt{k} = \texttt{3}$ and $\texttt{k} = \texttt{4}$) is required to continue the computation.

Suppose that in a computation of TM1 at some point $\texttt{n} \geq \texttt{3}$ and a right-moving IRR is needed to continue the computation. After this IRR has been applied, the length read is $\texttt{n}+\texttt{1}$ and suppose that $\texttt{k} < \texttt{4}$ for the next IRR. The movement resulting from applying this second IRR must also be to the right because the pointer starts at the right hand end of the read part of the tape and if it was to have a net left movement, $\texttt{k}$ would have to be the length of the read part of the tape prior to the second IRR i.e. $\texttt{n}+\texttt{1} \geq \texttt{4}$ because the

IRR would have to take the pointer to the left end of the tape. Alternatively if $\mathtt{k} \geq \mathtt{4}$, this must lead to a right movement as a result of the second IRR because all IRR($\mathtt{k}$) with $\mathtt{k} \geq \mathtt{4}$ are right-moving (Tables 2 and 3). Therefore the argument can be repeated indefinitely, showing that after a right-moving IRR is applied to a TM computation with TM1 with $\mathtt{n} \geq \mathtt{3}$, TM1 always goes to the right after each IRR is applied. These non-halting IRR must be of type RR and do not exist for length $\mathtt{k} > \mathtt{4}$ (they do not exist for $\mathtt{k} = \mathtt{5}$ and section 2.2 shows that they do not exist for any larger $\mathtt{k}$ either) therefore the TM moves after this using only the IRR of length $\leq \mathtt{4}$ i.e it moves only in a moving window of length 4 that moves one place to the right when a new symbol is read where the tape has not been read before.

It only remains to show its behaviour if $\mathtt{n} \geq \mathtt{3}$ and the TM continually takes left-moving IRR steps. These must be left-moving IRR($\mathtt{k}$) with $\mathtt{k} \leq \mathtt{3}$ of which only 5 for $\mathtt{k} = \mathtt{1}$ and 3 of type RL and 1 of type LL for $\mathtt{k} = \mathtt{2}$, and 3 of type RL for $\mathtt{k} = \mathtt{3}$ exist. Taking into account the rule types needed, only the 5 $\mathtt{k} = \mathtt{1}$ rules and the rule $\mathtt{4\underline{d}b} \rightarrow \mathtt{3\_ba}$ of type LL could be involved.

This implies that the TM moves in sequences of steps in IRR of lengths up to 2 only so it moves in a moving window of length 2. This behaviour could of course be permanently altered to a right-moving behaviour if at some point a right-moving IRR was required which of course depends on the next symbol read. Probably this is best characterised by the failure of the above conditions for a left-moving IRR.

3 Analysis of a TM simulating the Collatz problem

3.1 The origin of TM2.

The TM defined in this section is a trivial modification of one [2] chosen to simulate the Collatz [1] iteration. This reference is probably the most thorough introduction to it (there referred to as the $\mathtt{3x} + \mathtt{1}$ problem) containing the most important research papers on it, and is a pleasure to read because of the many connections with other fields and other fascinating simply stated but intractable conjectures.

This problem is easily stated as follows: For a positive integer $\mathtt{n}$, if $\mathtt{n}$ is even, replace it by $\mathtt{n}/\mathtt{2}$ and if $\mathtt{n}$ is odd, replace it by $\mathtt{3n} + \mathtt{1}$. It is well known that when this function is repeatedly applied to a positive integer the result seems to always eventually lead to 1, and then an infinite cycle 2,1,2, etc.. The problem can be stated formally (actually this is Terras's modification combining the $\mathtt{n} \rightarrow \mathtt{3n}+\mathtt{1}$ with $\mathtt{n} \rightarrow \mathtt{n}/\mathtt{2}$ because $\mathtt{3n}+\mathtt{1}$ is even when $\mathtt{n}$ is odd) as follows:

If the function $\texttt{f(n)}$ is defined from positive integers ($\mathbb{N}^+$) to themselves as follows:

$$\texttt{f(n)} = \begin{cases} \texttt{n}/2 \text{ if } \texttt{n} \text{ is even} \\ (3\texttt{n}+1)/2 \text{ if } \texttt{n} \text{ is odd} \end{cases} . \tag{53}$$

the problem is to show that

$$\forall \texttt{n} \in \mathbb{N}^+ \exists \texttt{k} \in \mathbb{N}^+ \text{ such that } \texttt{g}(\texttt{n},\texttt{k}) = 1 \tag{54}$$

where $\texttt{g}(\texttt{n},\texttt{k})$ is the $\texttt{k}$th iterate of $\texttt{f}$ applied to $\texttt{n}$ defined by

$$\forall \texttt{n},\texttt{k} \in \mathbb{N}^+ \texttt{g}(\texttt{n},\texttt{k}+1) = \texttt{f}(\texttt{g}(\texttt{n},\texttt{k})) \tag{55}$$

and $\texttt{g}(\texttt{n},1) = \texttt{f}(\texttt{n})$.

TM2 was obtained by changing Margenstern's TM (Figure 7 of [2]) thus: the symbols 0 and 1 were replaced by $\texttt{a}$ and $\texttt{b}$ respectively and the symbol $\_$ was replaced by $\texttt{c}$ and the quintuples left blank were replaced by halting computation rules (for $1\underline{\texttt{c}}$ and $2\underline{\texttt{c}}$). TM2 is defined by the following rules:

$1\underline{a} \to 1c\_$	$2\underline{a} \to 2a\_$	$3\underline{a} \to 2b\_$	$4\underline{a} \to 3a\_$	$5\underline{a} \to 5\_a$
$1\underline{b} \to 4c\_$	$2\underline{b} \to 3b\_$	$3\underline{b} \to 4a\_$	$4\underline{b} \to 4b\_$	$5\underline{b} \to 5\_b$
$1\underline{c} \to 1\underline{c}$	$2\underline{c} \to 2\underline{c}$	$3\underline{c} \to 5\_b$	$4\underline{c} \to 3a\_$	$5\underline{c} \to 1c\_$

Table 6: Rules of length 1 defining TM2 which simulates the $3\texttt{n}+1$ problem

In the following analysis, the same method used for TM1 is applied to TM2. This TM is a little more complex in some ways than the TM1 and a little more notation is needed to describe the IRR's. While developing this I have become aware that I have been developing a simple language for describing sets of rules, and more notations may be needed in future. The IRR's will be defined in terms of other sub-machines going in a single direction on the tape obtained directly from TM2 going either forward or backward. Braces will be used to indicate a set of symbols from which strings can be constructed e.g. $\{\texttt{a},\texttt{b}\}^\texttt{n}$ means the set of all strings of $\texttt{a}$'s and $\texttt{b}$'s, and the exponent $\texttt{n}$ indicates the length of the string, and the pointer indicated under the first or last brace indicates the pointer at the first or respectively last symbol in the string indicated by the expression in braces. Such strings can be embedded in others, and where one of several symbols or states is, the options will be listed vertically within large braces. Also the notation $|$ will be introduced to indicate the subset of the set indicated on the left with the condition indicated on the right. This subset will be the default argument in any functions. The name of any TM related to TM2 used as a function, will refer to that TM applied through as many cycles as possible to its argument. A related comment is appropriate here in relation to sets of CS's i.e. sets of sets of machine configurations. Without too much risk of ambiguity I have been using set theory terminology at the the upper level i.e. treating CS's as indivisible entities, which I will continue to do. However occasionally it may be necessary to work at the lower level for example talking about subsets or supersets of a CS.

3.2 Obtaining a recursive definition of the IRR for TM2

The IRR($\mathtt{n}$) for $\mathtt{n} = \mathtt{2}, \mathtt{3},$ and $\mathtt{4}$ are as follows respectively

$$\mathtt{3a\underline{c}} \to \mathtt{5\_ab} \quad \mathtt{3b\underline{c}} \to \mathtt{5\_bb} \quad \mathtt{5\underline{c}a} \to \mathtt{1cc\_} \quad \mathtt{5\underline{c}b} \to \mathtt{4cc\_} \tag{56}$$

$$\begin{array}{llll} \mathtt{3aa\underline{c}} \to \mathtt{5\_aab} & \mathtt{3ba\underline{c}} \to \mathtt{5\_bab} & \mathtt{3ca\underline{c}} \to \mathtt{4ccc\_} & \mathtt{3ab\underline{c}} \to \mathtt{5\_abb} \\ \mathtt{3bb\underline{c}} \to \mathtt{5\_bbb} & \mathtt{5\underline{c}aa} \to \mathtt{1ccc\_} & \mathtt{5\underline{c}ab} \to \mathtt{4ccc\_} & \mathtt{5\underline{c}ba} \to \mathtt{3cca\_} \\ \mathtt{5\underline{c}bb} \to \mathtt{4ccb\_} & & & \end{array} \tag{57}$$

$$\begin{array}{llll} \mathtt{3aaa\underline{c}} \to \mathtt{5\_aaab} & \mathtt{3baa\underline{c}} \to \mathtt{5\_baab} & \mathtt{3aba\underline{c}} \to \mathtt{5\_abab} & \mathtt{3bba\underline{c}} \to \mathtt{5\_bbab} \\ \mathtt{3cba\underline{c}} \to \mathtt{4ccaa\_} & \mathtt{3aab\underline{c}} \to \mathtt{5\_aabb} & \mathtt{3bab\underline{c}} \to \mathtt{5\_babb} & \mathtt{3abb\underline{c}} \to \mathtt{5\_abbb} \\ \mathtt{3bbb\underline{c}} \to \mathtt{5\_bbbb} & \mathtt{5\underline{c}aaa} \to \mathtt{1cccc\_} & \mathtt{5\underline{c}aab} \to \mathtt{4cccc\_} & \mathtt{5\underline{c}aba} \to \mathtt{3ccca\_} \;. \\ \mathtt{5\underline{c}abb} \to \mathtt{4cccb\_} & \mathtt{5\underline{c}baa} \to \mathtt{2ccab\_} & \mathtt{5\underline{c}bab} \to \mathtt{4ccaa\_} & \mathtt{5\underline{c}bba} \to \mathtt{3ccba\_} \\ \mathtt{5\underline{c}bbb} \to \mathtt{4ccbb\_} & & & \end{array} \tag{58}$$

A quick look at these rules as generated by the new program [5] for $\mathtt{n}$ up to 7 shows that every LHS of the form $\mathtt{5\underline{c}}\{\mathtt{a},\mathtt{b}\}^*$ and $\mathtt{3}\{\mathtt{a},\mathtt{b}\}^*\mathtt{\underline{c}}$ is present together with increasing numbers of LHS's of the form $\mathtt{3c}\{\mathtt{a},\mathtt{b}\}^*\mathtt{\underline{c}}$. The latter form about 1/7 of the LHS's having state $\mathtt{3}$ and occur precisely once every 7 rules listed in the way they are ordered. Also these rules are derived in 3 substitution steps and all the others are derived in just 2. Experience analysing small TM's this way shows that for $\mathtt{n}$ larger than about 5 or 6, the pattern of the IRR($\mathtt{n}$) for large $\mathtt{n}$ is clear, so I expect these statements to be true for all $\mathtt{n}$.

Directly from Table 6, the reverse rules of length 1 are

$$\left.\begin{array}{l}
\mathtt{1}\alpha_{_} \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{1}\underline{\mathtt{a}}, \mathtt{5}\underline{\mathtt{c}} \\
\mathtt{2}\alpha_{_} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{2}\underline{\mathtt{a}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3}\underline{\mathtt{a}} \end{cases} \\
\mathtt{3}\alpha_{_} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{4}\underline{\mathtt{a}}, \mathtt{4}\underline{\mathtt{c}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{2}\underline{\mathtt{b}} \end{cases} \\
\mathtt{4}\alpha_{_} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3}\underline{\mathtt{b}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{4}\underline{\mathtt{b}} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{1}\underline{\mathtt{b}} \end{cases} \\
\mathtt{5}\alpha_{_} \leftarrow \emptyset \\
\mathtt{st}_{_}\alpha \leftarrow \emptyset \text{ for } \mathtt{st} \in \{\mathtt{1},\mathtt{2},\mathtt{3},\mathtt{4}\} \\
\mathtt{5}_{_}\alpha \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{5}\underline{\mathtt{a}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3}\underline{\mathtt{c}}, \mathtt{5}\underline{\mathtt{b}} \end{cases}
\end{array}\right\} \text{ for } \alpha \in \{\mathtt{a},\mathtt{b},\mathtt{c}\} \tag{59}$$

Now adding single symbols at the pointer to these LHS's such that the forward computation generates an irreducible rule (i.e. a left moving step if the pointer is at the right and vice versa) yields just the 4 LHS's in (56) with their origins in (60).

$$\mathtt{3a}\underline{\mathtt{c}} \begin{cases} \leftarrow \mathtt{4}\underline{\mathtt{a}}\mathtt{c} \\ \leftarrow \mathtt{4}\underline{\mathtt{c}}\mathtt{c} \end{cases} \quad \mathtt{3b}\underline{\mathtt{c}} \leftarrow \mathtt{2}\underline{\mathtt{b}}\mathtt{c} \quad \mathtt{5}\underline{\mathtt{c}}\mathtt{a} \leftarrow \mathtt{5c}\underline{\mathtt{a}} \quad \mathtt{5}\underline{\mathtt{c}}\mathtt{b} \begin{cases} \leftarrow \mathtt{3c}\underline{\mathtt{c}} \\ \leftarrow \mathtt{5c}\underline{\mathtt{b}} \end{cases} \tag{60}$$

For the reverse rules of length 3, because all the rules in (56) are of type LR or RL they can all be extended by 1 symbol opposite the pointer giving irreducible rules with their origins (condition 1 only) thus:

$$\mathtt{3}\alpha\mathtt{a}\underline{\mathtt{c}} \begin{cases} \leftarrow \mathtt{4}\alpha\underline{\mathtt{a}}\mathtt{c} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3}\underline{\mathtt{b}}\mathtt{ac} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{4}\underline{\mathtt{b}}\mathtt{ac} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{1}\underline{\mathtt{b}}\mathtt{ac} \end{cases} \\ \leftarrow \mathtt{4}\alpha\underline{\mathtt{c}}\mathtt{c} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{3}\underline{\mathtt{b}}\mathtt{cc} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{4}\underline{\mathtt{b}}\mathtt{cc} \\ \overset{\alpha=\mathtt{c}}{\longleftarrow} \mathtt{1}\underline{\mathtt{b}}\mathtt{cc} \end{cases} \end{cases} \tag{61}$$

$$\mathtt{3}\alpha\mathtt{b}\underline{\mathtt{c}} \leftarrow \mathtt{2}\alpha\underline{\mathtt{b}}\mathtt{c} \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{2}\underline{\mathtt{a}}\mathtt{bc} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3}\underline{\mathtt{a}}\mathtt{bc} \end{cases} \tag{62}$$

$$\mathtt{5}\underline{\mathtt{c}}\mathtt{a}\alpha \leftarrow \mathtt{5c}\underline{\mathtt{a}}\alpha \begin{cases} \overset{\alpha=\mathtt{a}}{\longleftarrow} \mathtt{5ca}\underline{\mathtt{a}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{5ca}\underline{\mathtt{b}} \\ \overset{\alpha=\mathtt{b}}{\longleftarrow} \mathtt{3ca}\underline{\mathtt{c}} \end{cases} \tag{63}$$

$$\mathtt{5\underline{c}b}\alpha \leftarrow \mathtt{5c\underline{b}}\alpha \begin{cases} \overset{\alpha=\mathtt{a}}{\leftarrow} \mathtt{5cb\underline{a}} \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{5cb\underline{b}} \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{3cb\underline{c}} \end{cases} \tag{64}$$

This accounts for the list of 9 IRR(3) obtained from the program ($\alpha = \mathtt{a}, \mathtt{b}, \mathtt{c}$ from (61), $\alpha = \mathtt{a}, \mathtt{b}$ from (62), $\alpha = \mathtt{a}, \mathtt{b}$ from (63) and $\alpha = \mathtt{a}, \mathtt{b}$ from (64)). Extending the length of those that are of type RL or LR again by 1 in the same way gives results that can be summarised by the following sets of equations:

$$\left.\begin{array}{ll} \mathtt{3aaa\underline{c}} \leftarrow \mathtt{4\underline{\alpha}b}\beta\mathtt{c} & \mathtt{3bba\underline{c}} \leftarrow \mathtt{4\underline{b}b}\beta\mathtt{c} \\ \mathtt{3abb\underline{c}} \leftarrow \mathtt{4\underline{\alpha}abc} & \mathtt{3baa\underline{c}} \leftarrow \mathtt{2\underline{b}b}\beta\mathtt{c} \\ \mathtt{3aab\underline{c}} \leftarrow \mathtt{2\underline{a}abc} & \mathtt{3bbb\underline{c}} \leftarrow \mathtt{2\underline{b}abc} \\ \mathtt{3aba\underline{c}} \leftarrow \mathtt{3\underline{b}b}\beta\mathtt{c} & \mathtt{3bab\underline{c}} \leftarrow \mathtt{3\underline{a}abc} \end{array}\right\} \text{ for } \alpha, \beta \in \{\mathtt{a}, \mathtt{c}\}, \tag{65}$$

$$\mathtt{3cba\underline{c}} \leftarrow \mathtt{1\underline{b}b}\beta\mathtt{c} \text{ for } \beta \in \{\mathtt{a}, \mathtt{c}\}, \tag{66}$$

and

$$\mathtt{5\underline{c}}\alpha\beta\mathtt{a} \leftarrow \mathtt{5c}\alpha\beta\mathtt{\underline{a}} \quad \mathtt{5\underline{c}}\alpha\beta\mathtt{b} \begin{cases} \leftarrow \mathtt{5c}\alpha\beta\mathtt{\underline{b}} \\ \leftarrow \mathtt{3c}\alpha\beta\mathtt{\underline{c}} \end{cases} \text{ for } \alpha, \beta \in \{\mathtt{a}, \mathtt{b}\}. \tag{67}$$

Equations (65) (66) and (67) account for the 17 IRR(4) obtained from the program.

3.3 Setting up hypotheses to be proved by induction

For TM2 it appears to be true that

$$\mathtt{U_n} = \mathtt{3W_{n-1}\underline{c}} \tag{68}$$

plays an important role in its IRR where

$$\mathtt{W_n} = \{\mathtt{a}, \mathtt{b}\}^{\mathtt{n}} \tag{69}$$

i.e. any string of a's and b's of length $\mathtt{n}$. The origin of every member of $\mathtt{U_n}$ is obtained from $\mathtt{U_n}$ by using reverse TM steps in the same direction i.e. from the left. For each reversed computation path, this results in a CS in

$$\left\{\begin{array}{c} \mathtt{2} \\ \mathtt{3} \\ \mathtt{4} \end{array}\right\} \underline{\{}\mathtt{a}, \mathtt{b}, \mathtt{c}\}^{\mathtt{n}-1}\mathtt{c} \tag{70}$$

obtained from $\mathtt{U_n}$ by a repeated application of the reverse of TM2 (RTM2) restricted to states in $\{2, 3, 4\}$ and input symbols in $\{\mathtt{a}, \mathtt{b}\}$. RTM2 has a non-unique output in general and goes from right to left using Table 7 derivable from equations (59). The body of this table has the state-symbol pair(s) preceding the CS $\mathtt{S}\alpha_{\_}$ in TM2:

state	symbol α	
$\mathtt{S}$	$\mathtt{a}$	$\mathtt{b}$
$\mathtt{2}$	$\{\mathtt{2\underline{a}}\}$	$\{\mathtt{3\underline{a}}\}$
$\mathtt{3}$	$\{\mathtt{4\underline{a}}, \mathtt{4\underline{c}}\}$	$\{\mathtt{2\underline{b}}\}$
$\mathtt{4}$	$\{\mathtt{3\underline{b}}\}$	$\{\mathtt{4\underline{b}}\}$

Table 7: The definition of RTM2, the reverse of TM2 restricted to a subset of states and symbols

This situation will be indicated by

$$\mathtt{3W_{n-1}\underline{c}} \leftarrow \mathrm{RTM2}\left(\mathtt{3W_{n-1}\underline{c}}\right) \tag{71}$$

where RTM2 is used as a function to indicate the application of RTM2 through as many cycles as possible to its argument.

The LHS's of the IRR($\mathtt{n}$) appear to be the following forms: $\mathtt{3W_{n-1}\underline{c}}$, some subset of $\mathtt{3cW_{n-2}\underline{c}}$, and $\mathtt{5\underline{c}W_{n-1}}$. The calculation of the IRR(5) was carried out (not shown) which agreed with this and suggested the following forms for the LHS's of the IRR($\mathtt{n}$) together with their hypothesised origins:

$$\begin{aligned}
&\mathtt{U_n} \leftarrow \mathrm{RTM2}\left(\mathtt{U_n}\right) \text{ where } \mathtt{U_n} = \mathtt{3W_{n-1}\underline{c}} \\
&\mathtt{V_n}|_{\mathrm{S(RTM2(.))=4}} \leftarrow \left(\mathtt{4c_{_}} \leftarrow \mathtt{1\underline{b}}\right)\left(\mathrm{RTM2}\left(\mathtt{V_n}\right)|_{\mathrm{S(.)=4}}\right) \\
&\text{where } \mathtt{V_n} = \mathtt{3cW_{n-2}\underline{c}} \text{ and S means “state of”} \\
&\mathtt{5\underline{c}W_{n-2}}\alpha \begin{cases} \xleftarrow{\alpha\in\{\mathtt{a},\mathtt{b}\}} \mathtt{5cW_{n-2}}\underline{\alpha} \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{3cW_{n-2}\underline{c}} \end{cases}
\end{aligned} \tag{72}$$

The LHS of the second line can be expressed as the subset of $\mathtt{V_n} = \mathtt{3c}\{\mathtt{a},\mathtt{b}\}^{\mathtt{n-2}}\mathtt{\underline{c}}$ such that the state of RTM2($\mathtt{V_n}$) is $\mathtt{4}$ and its origin is the single reverse TM step $\mathtt{4c_{\_}} \leftarrow \mathtt{1\underline{b}}$ applied to the subset of RTM2($\mathtt{V_n}$) that has state $\mathtt{4}$. Note that although RTM2 can give many outputs for one input, the state of these is the same because this holds for a single reverse computation step in Table 7. It is straightforward to verify that (72) defines the IRR($\mathtt{n}$) for $\mathtt{n} = \mathtt{2}, \mathtt{3},$ and $\mathtt{4}$.

3.4 Proof of (72) by induction on n

Now apply the argument to extend the IRR to one extra symbol to start the proof of (72) by induction on $\mathtt{n}$. For (72).1 consider $\alpha\mathtt{U_n} = \mathtt{3}\alpha\mathtt{W_{n-1}\underline{c}}$ as possible

LHS's of IRR($\mathtt{n}+\mathtt{1}$). Their origins are the origins of αRTM2($\mathtt{U_n}$) which are all subsets of one of

$$\left\{\begin{array}{c}\mathtt{2}\\ \mathtt{3}\\ \mathtt{4}\end{array}\right\}\alpha_{\_} \tag{73}$$

whose origin can be traced back one further step using Table 7 (RTM2) provided $\alpha \in \{\mathtt{a},\mathtt{b}\}$ resulting in RTM2(αRTM2($\mathtt{U_n}$)) = RTM2($\alpha\mathtt{U_n}$). This results in termination under condition (1) of the procedure for tracing origins, so the new origin has been found. There are no other origins to be found by trying to go back in the other direction because the state is now $\mathtt{2}$, $\mathtt{3}$ or $\mathtt{4}$ and there is no possible preceding (in the sense of TM2) pointer position one place to the right because of (59).6

For the case $\alpha = \mathtt{c}$, using (59), an origin for $\mathtt{c}$RTM2($\mathtt{U_n}$) requires state $\mathtt{4}$ in (73). Therefore CS's of the form $\mathtt{3cW_{n-1}\underline{c}}$ are reachable if and only if RTM2 run to the leftmost but one symbol (i.e. as far as possible) results in state $\mathtt{4}$. Combining the results for $\alpha \in \{\mathtt{a},\mathtt{b}\}$ using $\mathtt{U_{n+1}} = \mathtt{aU_n} \cup \mathtt{bU_n}$ gives that $\mathtt{U_{n+1}}$ are reachable LHS's with origins obtainable via RTM2.

To check that these LHS's do all generate irreducible rules hence IRR's, note that the forward computation has the pointer passing every symbol in the string:

$$\mathtt{3}\alpha\{\mathtt{a},\mathtt{b}\}^{\mathtt{n}}\underline{\mathtt{c}} \to \mathtt{5}\alpha\{\mathtt{a},\underline{\mathtt{b}\}}^{\mathtt{n}}\mathtt{b} \to \mathtt{5}\underline{\alpha}\{\mathtt{a},\mathtt{b}\}^{\mathtt{n}} \to \dots \tag{74}$$

which follows simply from the machine table for TM2. This completes the proof of (72) parts 1 and 2 with length $\mathtt{n}+\mathtt{1}$. Equation (74) follows from the behaviour of state $\mathtt{5}$, which is simply to move to the left leaving the symbols unchanged provided they are all $\mathtt{a}$ or $\mathtt{b}$:

$$\mathtt{5}\{\mathtt{W_n}\underline{\}} \to \mathtt{5}\_\mathtt{W_n}. \tag{75}$$

This can be considered as behaviour of the trivial sub-machine of TM2 that has just state $\mathtt{5}$ and symbols $\mathtt{a}$ and $\mathtt{b}$. There is another behaviour of this sort that can be expressed by

$$\mathtt{1}\underline{\{}\mathtt{W_n}\} \to \text{TM2*}(\mathtt{1}\underline{\{}\mathtt{W_n}\})\_. \tag{76}$$

because starting with state $\mathtt{1}$ with symbols $\mathtt{a}$ and $\mathtt{b}$ on the right, TM2 can reach state $\mathtt{1}$,$\mathtt{2}$,$\mathtt{3}$, or $\mathtt{4}$ and continually go right unless a $\mathtt{c}$ is encountered. Here TM2* stands for TM2 restricted to the right-moving state and symbols combinations.

Next consider trying to extend the LHS's of the form in (72).2 to form LHS's of IRR($\mathtt{n}+\mathtt{1}$). This is easily proved impossible because, using (76) for the last step, the following is easily deduced:

$$\mathtt{3}\alpha\mathtt{cW_{n-2}\underline{c}} \to \mathtt{5}\alpha\mathtt{c}\{\mathtt{W_{n-2}}\underline{\}}\mathtt{b} \to \mathtt{5}\alpha\underline{\mathtt{c}}\mathtt{W_{n-2}b} \to \mathtt{1}\alpha\mathtt{c}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b} \to \mathtt{1}\alpha\mathtt{c}\text{TM2*}(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b})\_, \tag{77}$$

which shows that the pointer does not reach the symbol α, before reaching the RHS of the string of symbols, so the resulting rule is reducible.

It is easy to show that from (72).3 applying the same method to extend the reachability argument using (71) gives

$$\mathtt{5\underline{c}W_{n-2}}\alpha\beta \begin{cases} \xleftarrow{\alpha\in\{\mathtt{a,b}\}} \mathtt{5cW_{n-2}}\underline{\alpha}\beta \begin{cases} \xleftarrow{\beta\in\{\mathtt{a,b}\}} \mathtt{5cW_{n-2}}\alpha\underline{\beta} \\ \overset{\beta=\mathtt{b}}{\leftarrow} \mathtt{3cW_{n-2}}\alpha\underline{\mathtt{c}} \end{cases} \\ \overset{\alpha=\mathtt{b}}{\leftarrow} \mathtt{3cW_{n-2}\underline{c}}\beta \leftarrow \mathtt{c}\underline{\{}\mathtt{T_{n-1}}\}\beta \xleftarrow{\mathtt{S(T_{n-1})=4}} \mathtt{1\underline{b}T^*_{n-1}}\beta \end{cases} . \tag{78}$$

In this last equation, $\mathtt{T}^*$ indicates $\mathtt{T}$ with the state information deleted so that the new state can be given (in this case $\mathtt{1}$). Summarising this and deleting the bottom branch that ends in condition (2) gives the following

$$\mathtt{5\underline{c}W_{n-2}}\alpha\beta \begin{cases} \xleftarrow{\alpha \text{ and } \beta\in\{\mathtt{a,b}\}} \mathtt{5cW_{n-2}}\alpha\underline{\beta} \\ \xleftarrow{\alpha\in\{\mathtt{a,b}\} \text{ and } \beta=\mathtt{b}} \mathtt{3cW_{n-2}}\alpha\underline{\mathtt{c}} \end{cases} \tag{79}$$

i.e.

$$\mathtt{5\underline{c}W_{n-1}}\beta \begin{cases} \xleftarrow{\beta\in\{\mathtt{a,b}\}} \mathtt{5cW_{n-1}}\underline{\beta} \\ \overset{\beta=\mathtt{b}}{\leftarrow} \mathtt{3cW_{n-1}\underline{c}} \end{cases} \tag{80}$$

Checking for irreducibility of the rule using (76) gives

$$\mathtt{5\underline{c}W_{n-1}}\beta \rightarrow \mathtt{1c}\underline{\{}\mathtt{W_{n-1}}\}\beta \rightarrow \mathtt{c}\text{TM2}^*\left(\mathtt{1}\underline{\{}\mathtt{W_{n-1}}\}\right)\underline{\beta} \rightarrow \ldots \tag{81}$$

showing that the the rules generated from (72).3 by extension are irreducible for $\mathtt{n} > \mathtt{0}$ and therefore are IRR for $\beta \in \{\mathtt{a}, \mathtt{b}\}$. Note that because all possible extensions of the LHS's of the IRR($\mathtt{n}$) to one extra symbol have been considered, the set of LHS's of the IRR($\mathtt{n}+\mathtt{1}$) obtained here is complete on the assumption that the set of LHS's of the IRR($\mathtt{n}$) is complete. Thus (72) has been demonstrated with $\mathtt{n}$ replaced by $\mathtt{n}+\mathtt{1}$, and because (72) is true for $\mathtt{n} = 2$, (72) is true for all $\mathtt{n} \geq 2$ by induction.

Using the above results, it is now straightforward to derive the complete IRR($\mathtt{n}$) for TM2 as follows:

$$\begin{array}{l} \mathtt{3W_{n-1}\underline{c}} \rightarrow \mathtt{5}\{\mathtt{W_{n-1}}\underline{\}}\mathtt{b} \rightarrow \mathtt{5\_W_{n-1}b} \text{ [by (75)]} \\ \mathtt{5\underline{c}W_{n-1}} \rightarrow \mathtt{1c}\underline{\{}\mathtt{W_{n-1}}\} \rightarrow \mathtt{c}\text{TM2}^*(\mathtt{1}\underline{\{}\mathtt{W_{n-1}}\})_{\_} \text{ [by (76)]} \\ \left\{\mathtt{3cW_{n-2}\underline{c}} \rightarrow \mathtt{5\underline{c}W_{n-2}b} \rightarrow \mathtt{c}\text{TM2}^*(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b})_{\_}\right\}\Big|_{\text{S(RTM2(}\mathtt{3cW_{n-2}\underline{c}}\text{))=4}} \end{array} \tag{82}$$

where the result (82).3 follows from (82).1 and (82).2. These together with the original TM Table 6 characterise its behaviour. Note that the rules in (82).3 are valid without the restriction S(RTM2($\mathtt{3cW_{n-2}\underline{c}}$)) $= \mathtt{4}$ which only serves to remove redundant rules from the list.

3.5 Global analysis of TM2 based on its IRR and the Collatz conjecture

In an attempt to understand this TM further, the list of the IRR from (82).3 up to length 7 were obtained from the program and are as follows:

$$\begin{array}{lll}
\mathtt{3ca\underline{c}} \to \mathtt{4ccc_} & \mathtt{3cba\underline{c}} \to \mathtt{4ccaa_} & \mathtt{3caaa\underline{c}} \to \mathtt{4ccccc_} \\
\mathtt{3cbba\underline{c}} \to \mathtt{4ccbaa_} & \mathtt{3cabb\underline{c}} \to \mathtt{4cccbb_} & \mathtt{3cbaaa\underline{c}} \to \mathtt{3ccabab_} \\
\mathtt{3caaba\underline{c}} \to \mathtt{4ccccaa_} & \mathtt{3cbbba\underline{c}} \to \mathtt{4ccbbaa_} & \mathtt{3cabab\underline{c}} \to \mathtt{4cccaab_} \\
\mathtt{3cbabb\underline{c}} \to \mathtt{4ccaabb_} & \mathtt{3caaaaa\underline{c}} \to \mathtt{4ccccccc_} & \mathtt{3cbbaaa\underline{c}} \to \mathtt{3ccbabab_}\ . \\
\mathtt{3cabbaa\underline{c}} \to \mathtt{3cccbabb_} & \mathtt{3cbaaba\underline{c}} \to \mathtt{3ccabbbb_} & \mathtt{3caabba\underline{c}} \to \mathtt{4ccccbaa_} \\
\mathtt{3cbbbba\underline{c}} \to \mathtt{4ccbbbaa_} & \mathtt{3cabaab\underline{c}} \to \mathtt{4cccabba_} & \mathtt{3cbabab\underline{c}} \to \mathtt{4ccaaaab_} \\
\mathtt{3caaabb\underline{c}} \to \mathtt{4cccccbb_} & \mathtt{3cbbabb\underline{c}} \to \mathtt{4ccbaabb_} & \mathtt{3cabbbb\underline{c}} \to \mathtt{4cccbbbb_}
\end{array} \tag{83}$$

These are useful for hand calculations with TM2.

The behaviour of TM2 based on (82) can be described as follows. If TM2 starts from $\mathtt{3\underline{c}}$ it enters state $\mathtt{5}$ going left over $\mathtt{a}$'s and $\mathtt{b}$'s, which are left unchanged ((82).1). Then if a $\mathtt{c}$ is encountered (82).2 applies leaving that $\mathtt{c}$ in place. This $\mathtt{c}$ reverses the TM and takes it to state $\mathtt{1}$, after which the next symbol printed must also be a $\mathtt{c}$. TM2 then continues in TM2* mode and going to the right. TM2* mode is entered from state $\mathtt{1}$, and state $\mathtt{1}$ is maintained only as long as $\mathtt{a}$'s are present and shifting to state $\mathtt{4}$ as soon as the first $\mathtt{b}$ is encountered. $\mathtt{c}$'s are being printed at this time so the configuration is here in $\mathtt{4c\underline{a}}$ or $\mathtt{4c\underline{b}}$. Note that TM2* cannot print a $\mathtt{c}$ unless it is in state $\mathtt{1}$ or $\mathtt{5}$ which cannot be reached from other states within TM2*. If a $\mathtt{c}$ is again encountered in state 3 a cycle has been completed because $\mathtt{3\underline{c}}$ is then reentered (and a $\mathtt{c}$ is required to get TM2 out of TM2* mode). The $\mathtt{c}$ cannot be encountered in state $\mathtt{1}$ if at least one $\mathtt{b}$ is present in $\mathtt{W_n}$ and it cannot be encountered in state 2 because by (82).3 the last symbol in the string to be processed by TM2* is a $\mathtt{b}$, so the state entered after this must be $\mathtt{3}$ or $\mathtt{4}$. Therefore the cases when the TM effectively halts (remains perpetually stationary) cannot occur when starting from any configuration in $\mathtt{3cW_{n-2}\underline{c}}$. If a $\mathtt{c}$ is encountered in state $\mathtt{3}$, TM2 completes the cycle, and if a $\mathtt{c}$ is encountered in state $\mathtt{4}$, one extra TM step with input symbol $\mathtt{c}$ will bring it back to a configuration in $\mathtt{3\underline{c}}$. If the input symbol here is not a $\mathtt{c}$ the computation remains in TM2* mode searching for the next $\mathtt{c}$'s. For TM2, $\mathtt{c}$'s can be created or deleted going right, but going left in state $\mathtt{5}$ a $\mathtt{c}$ results in two successive $\mathtt{c}$'s being printed, hence in general TM2's behaviour is to move the leftmost point in its cycle to the right by at least one place per cycle, in cycles of arbitrary length.

The cycle is in fact an almost completely general description because every state-symbol pair can be included somewhere in the cycle, $\mathtt{5\underline{a}}$ and $\mathtt{5\underline{b}}$ in the left-moving part, and $\{\mathtt{1},\mathtt{2},\mathtt{3},\mathtt{4}\} \times \{\mathtt{\underline{a}},\mathtt{\underline{b}}\} \cup \{\mathtt{4},\mathtt{5}\} \times \{\mathtt{\underline{c}}\}$ in the right-moving part, and all the other cases with symbol $\mathtt{c}$ are included too. Thus whatever the starting condition of the TM, it can be mapped onto a point in this cycle. This behaviour is simple enough that it could have been obtained directly from the TM in this example. The exceptions implied above are (1) when in TM2* mode the machine happens to reach a $\mathtt{c}$ in state 2 when it halts and (2) when it reaches a $\mathtt{c}$ in state 1 leading to a halt. The latter can only result from $\mathtt{1\underline{a}c}$ and $\mathtt{5\underline{c}c}$. These cannot happen as part of the cycle described above and were found by manual calculation, so the algorithm described above is ignoring these somewhat trivial cases.

This analysis is somewhat paradoxical in that previous analyses have always led to a systematic "unravelling" of the effect of back and forth movements of the TM, but in this example so far, the analysis has led to just the description of a cycle rather than summarising its total effect, which seems to be because cycles of arbitrary length can occur for the TM. Another clue to the fact that something very different is happening in this example is that a subset of the set of IRR deducible from the the IRR for the previous value of $\mathtt{n}$ are included i.e. (82).3.

For an analysis of any TM that can simulate any other one, the description of its interpretive cycle must *be* its analysis in general because anything else will depend on the TM being simulated, and because this is a cycle, in *general* any analysis beyond a cyclic description cannot be obtained. This suggests that any further analysis with this example must be analysis of special cases rather than the general theory above, for example analysis starting from CS's of the form

$$\mathtt{3c}^\infty\{\mathtt{W_{n-1}}\}\underline{\mathtt{c}}\mathtt{c}^\infty \tag{84}$$

where $\mathtt{c}^\infty$ means an infinite string of $\mathtt{c}$'s. The choice (84) was made suggested by the LHS of (82).3 and its close correspondence with

$$\mathtt{1c}^\infty\underline{\{}\mathtt{W_{n-1}}\}\mathtt{bc}^\infty \tag{85}$$

via (82).1 and the first step of (82).2, but moreover (85) are the CS's describing the initial configuration defined in [2] in which a number is given as input to TM2 in reversed binary i.e. with the most significant digits at the right, where $\mathtt{a}$ and $\mathtt{b}$ represent 0 and 1 respectively, and the $\mathtt{c}$'s represent blanks. The effect of the $\mathtt{c}^\infty$ at each end of the configuration will be implemented simply by adding an extra $\mathtt{c}$ at the pointer whenever it goes beyond the existing symbols. Extra $\mathtt{c}$'s at the ends may be dropped or added arbitrarily to the right hand side and $\rightarrow$ will be written like $\xrightarrow{\mathtt{c}}$ to take on this meaning.

To show that this can be repeated indefinitely, what needs to checked is that (1) the RHS of (82).3, with the extra step done if needed ($\mathtt{T}$) is again of the form of the LHS of (82).3 and (2) the extra condition in (82).3 is always satisfied i.e. $\mathrm{S}(\mathrm{RTM2}(\mathtt{T})) = \mathtt{4}$.

Start by finding the most explicit expressions for $\mathtt{T}$. If $\mathtt{X} = \mathtt{c}\mathrm{TM2}^*(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b})\underline{\mathtt{c}}$ is in state $\mathtt{3}$ then $\mathtt{T} = \mathtt{X}$ and if $\mathtt{X}$ is in state $\mathtt{4}$, $\mathtt{T}$ is obtained from $\mathtt{X}$ by going forward one step to get $\mathtt{T} \stackrel{\mathtt{c}}{=} \mathtt{c}\mathrm{TM2}^*(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b})\mathtt{a}\underline{\mathtt{c}}$ not forgetting that c's need to be added at the pointer. Here I introduce the notation $\stackrel{\mathtt{c}}{=}$ to indicate equality with an arbitrary number of c's added or subtracted from either end of the string. Thus

$$\mathtt{T} \stackrel{\mathtt{c}}{=} \mathtt{c}\mathrm{TM2}^*\left(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b}\right)\left\{\begin{array}{c}\underline{\mathtt{c}}\\ \mathtt{a}\underline{\mathtt{c}}\end{array}\right\} = \mathtt{f}(\mathtt{c}\mathrm{TM2}^*\left(\mathtt{1}\underline{\{}\mathtt{W_{n-2}}\}\mathtt{b}\right)), \tag{86}$$

depending on whether the state was 3 or 4 respectively at the penultimate step and f indicates the final step indicated above if required. The state of T is 3. Separating out the first row of a's, one can write

$$\mathtt{W_{n-2}} = \left\{\begin{array}{l}\mathtt{a^k b W^*_{n-k-3}} \text{ for some } \mathtt{k} \text{ such that } \mathtt{0} \leq \mathtt{k} \leq \mathtt{n-3} \quad \text{or}\\ \mathtt{a^{n-2}}\end{array}\right. . \tag{87}$$

Assuming first (87).1 the typical case, notice that state 1 is maintained under TM2* only as long as a's are present. After the first b, state 4 is entered so

$$\mathtt{T} \stackrel{\mathtt{c}}{=} \mathtt{f}(\mathtt{c}\mathrm{TM2}^*\left(\mathtt{4}\underline{\{}\mathtt{W^*}\}\mathtt{b}\right)). \tag{88}$$

Only a and b can be written in this application of TM2* after this point. State 1 cannot be reentered in this application of TM2* and the b at the end shows that the state after TM2* has finished is 3 or 4, and f finally takes it to 3. Here $\mathtt{W^*}$ is another string $\in \{\mathtt{a},\mathtt{b}\}^*$. Notice that here c's have been deleted from the left hand end of the string. Therefore T is indeed of the form $\mathtt{3cW_{n-2}}\underline{\mathtt{c}}$ where $\mathtt{W_{n-2}}$ is not necessarily the same as given by (87). If however (87).2 holds then, provided $\mathtt{n} \geq \mathtt{2}$,

$$\mathtt{T} = \mathtt{f}(\mathtt{c}\mathrm{TM2}^*\left(\mathtt{1}\underline{\{}\mathtt{a^{n-2}}\}\mathtt{b}\right)) \quad \stackrel{\mathtt{c}}{=} \mathtt{f}(\mathtt{4c^{n-1}}\underline{\mathtt{c}}) \stackrel{\mathtt{c}}{=} \mathtt{3ca}\underline{\mathtt{c}} \tag{89}$$

This result is again of the form $\mathtt{3cW_{n-2}}\underline{\mathtt{c}}$ and completes part (1) of the proof. Returning to (2), it must be shown that the state of RTM2(T) = 4. RTM2 takes the CS back to what it was before, except that c's can appear where a's were before because $\mathtt{3a\_} \leftarrow \left\{\begin{array}{c}\mathtt{4}\underline{\mathtt{a}}\\ \mathtt{4}\underline{\mathtt{c}}\end{array}\right.$ and this does not affect anything else because the state is not dependent on which of these different actions in RTM2 is taken. The c on the left stops RTM2 because it gets back to $\mathtt{4c}\underline{\mathtt{a}}$ or $\mathtt{4c}\underline{\mathtt{b}}$ that come from $\mathtt{1}\underline{\mathtt{b}}$, but state 1 is not included in RTM2, so RTM2 stops before going back this step and stops in state 4 as required. In case (87).2, RTM2 undoes the final step of (89) leaving it in state 4. This completes the proof of the following theorem

Theorem 3.1. *For computation with TM2 starting from a CS of the form* (84)*, the following cycle can be repeated indefinitely: (1) Use of a member* *of* (82)*.3 to make a substitution followed by (2) a single TM step after the inclusion of a symbol* $\texttt{c}$ *at the pointer. Step (2) is to be implemented only if the result of step (1) was in state* $\texttt{4}$*.*

An example of this is as follows starting from the representation of the number 27 just showing the CS's in state $\texttt{3}$:

$$
\begin{array}{l}
\texttt{3cbbabb}\underline{\texttt{c}}\\
\texttt{3ccbaabba}\underline{\texttt{c}}\\
\texttt{3cccabbaaaa}\underline{\texttt{c}}\\
\texttt{3cccccbabaab}\underline{\texttt{c}}\\
\texttt{3ccccccaaabbaa}\underline{\texttt{c}}\\
\texttt{3ccccccccccbabb}\underline{\texttt{c}}\\
\texttt{3cccccccccccaabba}\underline{\texttt{c}}\\
\texttt{3ccccccccccccccbaaa}\underline{\texttt{c}}\\
\texttt{3cccccccccccccccabab}\underline{\texttt{c}}\\
\texttt{3cccccccccccccccccaaba}\underline{\texttt{c}}\\
\texttt{3ccccccccccccccccccccaaa}\underline{\texttt{c}}\\
\texttt{3cccccccccccccccccccccccca}\underline{\texttt{c}}
\end{array}
\tag{90}
$$

which is easy to derive using (83) extended to length 9. Thereafter it is easy to show that the pattern just moves to the right because

$$\texttt{3ca}\underline{\texttt{c}} \to \texttt{5}\underline{\texttt{c}}\texttt{ab} \to \texttt{1c}\underline{\texttt{a}}\texttt{b} \to \texttt{1cc}\underline{\texttt{b}} \xrightarrow{\texttt{c}} \texttt{4ccc}\underline{\texttt{c}} \xrightarrow{\texttt{c}} \texttt{3ccca}\underline{\texttt{c}}. \tag{91}$$

The obvious question is to determine whether the following statement is true or not:

Hypothesis 1. *Every computation of TM2 starting from a configuration in* (84) *leads to the cycle defined by* (91)*.*

When executing the cycle in Theorem (3.1) using (82).1 and (82).2, (82).1 does not alter any symbol between the two $\texttt{c}$'s, but then $\texttt{5}\underline{\texttt{c}} \to \texttt{1c}_{\_}$, $\texttt{1}\underline{\texttt{a}} \to \texttt{1c}_{\_}$ and $\texttt{1}\underline{\texttt{b}} \to \texttt{4c}_{\_}$ ensure that the existing $\texttt{c}$ at the left is kept and one more is added to its right, so that the number of $\texttt{c}$'s at the left increases by at least one per cycle. Actually because $\texttt{5}\underline{\texttt{c}}\texttt{a}^{\texttt{n}}\texttt{b} \to \texttt{4c}^{\texttt{n+2}}{}_{\_}$, the number of $\texttt{c}$'s at the left must in general increase at a rate of more than one per cycle. In addition, the information in the $\texttt{W}_\texttt{n}$ must move by just one space per cycle given by TM2*, therefore eventually all the information in the $\texttt{W}_\texttt{n}$ would be erased as the $\texttt{c}$'s advance from the left if it were not for the effect of the state, which when equal to 4 behaves a bit like a carry and has the effect of putting an extra $\texttt{a}$ on the tape before returning to state 3. Using this idea, if iterations like (90) are split into chunks where all the previous "information" has just been deleted, and new "information" resulting from the "carries" appears, a map exists describing this. If some property of this information defined by a non-negative integer strictly decreases as a result of the map and is zero only in the case (91) then this would establish Hypothesis 1.

Hypothesis 1 implies that the corresponding sequence of numbers defined by the iteration of (53) in the reverse binary form in $\mathtt{W_n}$ in state `1` always eventually get to 1 and 2 alternating indefinitely, because any starting configuration of the form (85) corresponding to any initial value in the sequence will give rise to a configuration in (84) as a result of part of the cycle defined in Theorem 3.1, which then leads to the behaviour in (91) which indicates in state `1` the numbers 2 and 1 in reverse binary notation. Therefore Hypothesis 1 implies that the Collatz conjecture is true.

To end on a philosophical point, referring to page 20 of [1], I don't think it makes sense to say that a single yes/no question such as the Collatz conjecture is unsolvable. Provability within a formal system is another matter. This requires the proof to be of the type that can be stated formally within the system. Proofs can sometimes be made outside the system for statements that cannot be proved within it, such as in the proof of Gödels first incompleteness theorem. The following statement is I think a matter of faith (to which I adhere), i.e. not capable of proof: "Every true statement of mathematics has a proof" thus the Collatz conjecture can be definitely proved true or false and presumably eventually a proof will be found.

References

[1] J.C. Lagarias (Editor), The ultimate challenge: The 3x+1 problem, The American Mathematical Society(Providence RI U.S.A., 2010).

[2] M. Margenstern, Theoretical Computer Science, 231(2000),217-251.

[3] J.H. Nixon, Mathematica Aeterna, 3(2013),709-738.

[4] J.H. Nixon, A computer program written in C++ for the analysis of Turing Machines

[5] J.H. Nixon, The updated version in D of the computer program for analysis of Turing Machines

Received: January 11, 2017