Exploring Halting Times for Unconventional Halting Schemes

Katarzyna Krzyzanska
Wolfram Summer Camp
East Amherst, NY 14051, United States of America
18kkrzyzanska@nardin.org

A common issue in the study of Turing machines (TMs) is the halting problem, or whether and when a TM will cease moving. Generally, this problem has been proved to be uncomputable, though it is possible to determine halting probabilities for more specific cases. In the following study, halting probabilities were determined using two unconventional definitions of halting. The first defines halting as the point where a machine reaches a certain, prespecified step; the second defines halting as the point where cell states stop changing (though head states may still differ from step to step). Due to computational limitations, the halting probabilities for TMs with fewer head and cell states have been more thoroughly studied than for more complicated machines, but nonetheless some data has been garnered concerning those. The TMs studied ranged from two possible head states and two possible cell states to six possible head states and six possible cell states.

Keywords: Turing machines; halting problem; complexity; estimation

1. Introduction

A Turing machine (TM) is a complex system where, on a row of cells, the direction of the "head" and the state of the current cell determine what the next head and cell states will be, taking into account the movement of the head left or right on the row of cells.

For example, a head state of one (perhaps indicating that the head is facing up) with a cell state of zero (perhaps indicating a blank cell) may follow such a rule that the head would next change to a state of two, change the cell state to one, and move one unit to the left. This can be represented as $\{1, 0\} \to \{2, 1, -1\}$.

A set of rules expressing all possible head/cell state combinations may appear as shown in Figure 1.

The number of possible head and cell states is chosen as convenience allows. The simplest useful TMs have two possible cell states and two possible head states, and are thus denoted as 2,2 TMs. There are 4096 such machines, given by the formula $(2sk)^{sk}$ [1, 2], where s

https://doi.org/10.25088/ComplexSystems.27.1.85

is the number of possible head states and k is the number of possible cell states. As can be easily noted, the number of possible TM rules increases at a great rate, so although TMs reaching up to 6,6 TMs have been studied, the amount of data that would have to be analyzed in order to obtain meaningful results has slightly limited the extent to which these and other more complicated TMs could be analyzed.

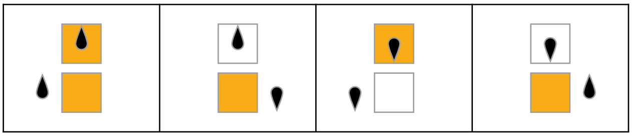


Figure 1. Example rule plot of a 2,2 TM given by Mathematica. This particular plot exemplifies rule 1507.

A TM will generally either halt relatively quickly or never halt, as proven by Calude and Stay [3], though there exist a few exceptions where a TM will only halt after an "algorithmically compressible" amount of time [4]. Within this paper, not only is this theoretical expectation examined numerically, but also the problem of reachability is explored, both in this context and that of a fixed, unchanging point of the machine tape. Granted, the issue of determining whether the tape indeed reaches this fixed point is undecidable as a general case due to the halting problem, but experimentally it was found that for small, specific samplings it was possible to determine. This will be further discussed in Section 3. The time at which a TM halts is a function of its algorithmic (or Kolmogorov–Chaitin) complexity [1, 3–6], and it is known that with time, the halting probability of a TM that has not halted yet approaches zero [4, 7].

It should be noted that the TMs used in this project have been enumerated using the Wolfram programming language, and as such have no specific halting state. Therefore, this paper concerns TMs with an arbitrary halting configuration. The two halting configurations studied include a halt state corresponding to some head/cell state combination and the aforementioned unchanging point in the tape. That is, per the second definition of halting, a TM halts when it reaches a point where the cell state no longer changes. It is in this sense that this study varies from others such as [5] and [7]. Furthermore, neither does this study use TMs in the context of axioms as [7] does, nor are the direct outputs of TMs analyzed as in [5]. Though all three of these studies utilized Mathematica, they all made use of it for different purposes (the use of *Waldmeister* in [7], for example), some of which may have contributed to differences in halting time distributions. This issue, as well as other such dissimilarities, is touched upon in Section 2.1. Additionally, the scope of this exploration differs from that of either [5] or [7], as those largely discuss n,2 TMs where $0 < n < 5$, whereas this study obtained results for s,k TMs reaching $1 < s < 7$ and $1 < k < 7$ through random selection (s,k values of one were assumed trivial and values of seven or more were too great to reasonably compute).

Since for certain TMs maximum halting times are known, any such machine that does not halt before the maximum time will not halt after it [7]. These known maximum halting times will be used in the first definition of halting.

For the purposes of this study, every TM is assumed to have an infinite, one-dimensional, blank (all cell states are zero) tape at its disposal, and the range that the head moves from one step to another has been limited to one unit left or right. The head is assumed to start at a state of one. This differs from other configurations where, for example, a one-sided tape with a finite right end can define halting simply by when the TM reaches this end [1, 8].

2. Halting as Defined by Rules

As aforementioned, a TM will not halt unless some halt state is specified. In this instance, the halt state for a given TM is defined within the set of rules in a manner similar to busy beaver TMs. Busy beavers are machines with n head states and two cell states (denoted n,2) that attempt to achieve the maximum number of steps before halting. These are known to have maximum halt values of six, 24 and 107 steps for two, three and four possible head states, respectively [7, 9].

The rules can be defined by assigning every head state (s) on a cell of state (a) to a new head state (sp), cell state (ap) and head displacement (off), denoted $\{s, a\} \to \{sp, ap, off\}$. Here, the halt state was indicated by assigning a single $\{s, a\}$ in any set of rules to $\{0, 3, 0\}$. In practice, this did not actually halt the TM, but it did provide a method to determine when a machine would have halted. As no other rule would result in a head state of zero or a cell state of three, it was easy to identify computationally which machines "halted."

Definition 1. A Turing machine has halted when it reaches a head/cell state combination that, in its rule set, halts the machine.

2.1 Halting Probabilities of 2,2 Turing Machines

Using functions created within Mathematica, all possible TMs with two possible cell states and two possible head states in addition to the halting state were enumerated, and the percent of those halted was calculated, giving the approximate halting probability for 2,2 TMs. This code and all code that follows in this paper was created with Mathematica and is available at https://github.com/kk428/TM-NCHalts/blob/master/TM_Code_Packge.m.

When these functions were run, it was shown that any given 2,2 TM with a halting state defined in its rules has about a 43.3% chance of halting. This is markedly higher than the probability given by [5], about 30.44%, and [7], about 34.56%. It is lower than the probability given by [1], which is about 66.7%, though this was on a one-sided tape. It should be kept in mind, furthermore, that the enumeration of halting machines differs in this paper.

The total number of rules that halt at a certain step was likewise given by a function. It was found that 2016 TMs halted at the first step, 1008 at the second, 288 at the third, 108 at the fourth, 12 at the fifth and 60 at the sixth.

Figure 2 plots the step at which these TMs halt against their frequency. For the most part, there is a great downward trend concerning at what step a 2,2 TM halts. It is far more likely for a machine to halt at the first step than at the sixth, which makes sense. The TMs that halted at the first step would have to consist of all those whose rules dictated that a head state of one and a cell state of zero calls for a halt. Any TM with this as part of its rule set will definitely halt on step 1. On the other hand, those rules that halt on step 2 have to depend on two factors: the rule dictating what should happen after the initial state and the rule dictating what should be done after that. It would follow that this compounding complexity results in fewer instances of halting. And of course, each step results in more complications, leading to a progressively decreasing halting probability.

The plot given in Figure 2 approximates the Levin universal semimeasure distribution (see [5, 6]), though it may vary from it somewhat because of how the halting state was defined and how TMs were enumerated. Other approximations given by the distributions in [5] and [7], for example, both employed 10 000 2,2 TMs using the formula $(4n+2)^{2n}$ for n,2 TMs. This study instead found 4096 2,2 TMs given by $(2sk)^{sk}$ for s,k TMs, as did [1], though this does not account for the additionally defined halting state. However, taking that into account, there would still only be a total of 8064 TMs per this study's creation of TM rules. This discrepancy could clearly contribute to differences in probability and intrinsically implies a disparity in TM enumeration. Additionally, although this study, [5] and [7] all used busy beavers in a sense to define halting (where there is a halt state of zero outside of the two head and cell states), here the halting state had to be explicitly implemented into the TM rules within the Mathematica TuringMachine function, and thus the TM rules themselves were generated as systematic combinations, further affecting enumeration. (Though [5] likewise used the TuringMachine function, it can be assumed that the formulation differed.)

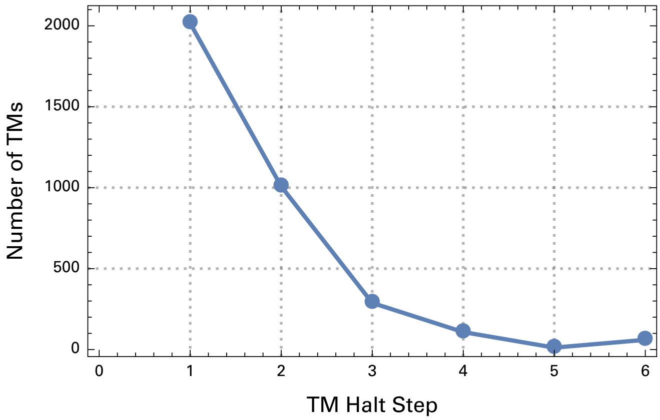


Figure 2. Plot of the step at which a function halts against how many functions halt at that step. The data points have only been joined for visualization purposes. Note that this plot is fairly similar to Figures 1 and 6 in [7] in terms of values and trends, respectively, except for the value of the number of TMs that halt at five steps (12 compared with 362) in Figure 1. Though Figure 2 more or less follows the trend given by Table 2 of [5], the differences in the result measured (string length rather than halting time) allow for no more than an estimate. Here there appear to be five times as many TMs halting at the sixth step as at the fifth; it is assumed to be a random result.

2.2 Halting Probabilities of 3,2 Turing Machines

Theoretically, the preceding process could be performed for 3,2 and 4,2 TMs as well. However, the runtime needed to run the 3,2 TM counterpart to the aforementioned functions as well as the memory space it requires made it unreasonable to proceed. It could be imagined that functions dealing with 4,2 TMs would create even greater difficulties.

Alternatively, it is possible to randomly select 3,2 TMs in order to at least obtain an approximation of the halting probability. Therefore, a function that randomly generates a rule set and determines whether it halts was defined.

This function resulted in, when run to select 1000 TMs five times, 609, 575, 576, 593 and 602 nonhalting values and thus 391, 425, 424, 407 and 398 halting values. Runtime generally limited the creation of a larger sample set. Taking the average of these values gives an approximate halting probability of about 40.9% for 3,2 TMs.

Although this value is fairly close to the 43.3% halting probability of 2,2 TMs, as not a single value exceeded or matched it among the non-averaged values, it seems fair to assume that the halting probability decreased from 2,2 to 3,2 TMs.

This could be because even though 3,2 TMs have more head/cell state combinations, for every rule set there is still only one head/cell state that results in a halt state. For a given rule set, all head/cell state combinations may not have the same probability of occurring within a TM, but it follows that a randomly selected TM would be less likely to generate the precise head/cell combination needed at a given step rather than one of the many other possible combinations. If this pattern continues, then it could be said that the total possible TMs increase at a faster rate than halting TMs. In this case, by increasing possible head or cell states, the TM becomes overall less likely to halt in the style of the increasing complexity of 2,2 TMs.

2.3 Halting Probabilities of 4,2 Turing Machines and *n*,2 Turing Machines with Greater Numbers of Head States

In the same way as used for 3,2 TMs, the halting probabilities of 4,2 TMs can be, in theory, determined using random samples. However, the maximum halting times for n,2 TMs where n is greater than four are unknown, and furthermore, even the lower limits determined for certain values of n are too large to effectively use in computations. For example, 5,2 TMs have a lower limit of 47 176 870 maximum steps, whereas 4,2 TMs have only 107 [1, 5, 9].

Concerning the aforementioned maximum halt times, the only time they would be used in calculations would be to provide a limit for the halting state to develop. It could be reasonably possible to gather some data within the first 10 000 steps or so of 5,2 TMs, for example, but this may not necessarily represent the overall halting probability.

However, even though it does seem possible to approximate halting probabilities for 4,2 TMs, even this seems to be unfeasible, or at least difficult, in terms of computational strength (see [1, 8] and especially [5]). The great number of rule sets to choose from (rather than the maximum halting time) restrains the selected computations from generating even one random 4,2 TM, let alone enough to make up a sample size. Though Section 5 of [1] referred to the random selection of 4,2 TMs, it should be noted that both the sampling process and how halting was defined differed somewhat. The sampling process included a correlation with 3,2 TMs, and TM halting itself was treated as the completion of a function rather than the definition used previously. Furthermore, even in this case it was acknowledged that the samples garnered may be "not at all representative."

2.4 Halting Probabilities of 2,3 Turing Machines

In the same vein as the problems of 5,2 TMs and greater, most n,3 TMs have unknown maximum halting times or ones that are too great to produce meaningful results. Only 2,3 TMs with a maximum halting time of 38 are within the range of reasonable computation; even 3,3 TMs have a lower limit of 374 676 383 maximum steps [10]. Selecting random halting TMs is therefore an effective method to approximate the halting probability of 2,3 TMs.

Running a function analogous to the one mentioned in Section 2.2, values of 680, 665, 686, 673 and 670 total nonhalting TMs were returned, respectively giving values of 320, 335, 314, 327 and 330 halting TMs. The average of these values yields a halting probability of 32.52% for 2,3 TMs.

This, much like in Section 2.2, could be explained by increasing TM complexity. As increasing the possible cell states by one decreased the halting probability by significantly more than increasing the possible head states by one did (−10.78% instead of −2.4%), it appears that here cell states may affect TM functionality to a greater degree than head states do.

3. Halting as Defined by Cell State

An alternative way to define a halting state could be to claim that a machine halts when the cell state no longer changes from step to step after reaching a fixed point. This, however, allows for the head to keep changing position. It has been noted that the general case for deciding whether a given TM eventually reaches this fixed point is undecidable because of the halting problem. However, here only small, specific samples of TMs were tested using only 100 steps to determine whether they eventually no longer vary in cell state. Though 100 may seem a bit low considering the lower limits on potential busy beavers, it was found that probability values did not vary significantly between using 100 and 1000 values, and furthermore that using even greater values would be computationally unfeasible. In this case, to determine whether a TM without a halt state explicitly defined in its rules will eventually reach this unchanging point, a function was created that looks for four repeats in a row at $n-1$, n, $n+1$ and $n+2$ steps, then determines whether this pattern holds true at $2n$. Here, 100 was used as n. If a TM held to the same cell state for such a length of time, it can be assumed that the cell states would no longer change.

Definition 2. A Turing machine has halted when there is no longer a change in cell states from step n to step $n+1$ for any value of n above a given limit.

3.1 Explanation of the Need for $2n$

Originally, the pattern at $2n$ (or even at $n+2$) was not evaluated, as it seemed natural that a TM that produces the same result three times in a row would continue to do so for the fourth, fifth, sixth and so forth time. However, the fact that the head moves and may be in different positions even when cell states are identical had been underestimated, leading to a mistake in computations noted when the halt times of various TMs were plotted and outliers were uncovered that could only be accounted for by the fact that these TMs did not actually halt.

Most of the values, predictably, required fewer than six steps to halt, but three data points appeared to have halted after 90 steps, indicating a mistake in defining halting. These TMs simply had repeating sets of three cell states throughout.

By additionally checking for $2n$ and $n+2$, it was ensured that a given TM actually does have constant, identical cell states from a certain point onward. It may be conceivable that a function repeats four times, continues erratically, and then by chance hits upon the same row of cell states as before at $2n$, but this seems highly improbable. Even if a significant number of TMs have four-step repetition, it is unlikely that this combination of cell states would occur again at $2n$. Furthermore, the three outliers detected previously indeed had three-step repetition, but each cycle of three steps differed from the previous one (though, admittedly they do not perhaps exemplify an adequate sample set). At any rate, this exception seems unlikely enough to be discounted in random selection.

3.2 Halting Probability for Various Turing Machines

To create a sample set, a function was created that generates 1000 random TMs within a certain set of s,k TMs where the rules were predefined by Mathematica as integer values ranging from zero to $\left((2sk)^{sk}\right)-1\right)$. Each of these TMs is run for 100 steps and is classified as halting or nonhalting according to the aforementioned method.

With these values, a table of halting probabilities for TMs ranging from 2,2 to 6,6 TMs was created (see Table 1).

This data can be used to observe trends both when the cell states keep constant while the head states change and when the head states are constant and the cell states change, given by Figures 3 and 4, respectively.

The cell state appears to determine halting probabilities to a greater degree than the head state, much as was noted in Section 2.4. However, perhaps in this case this results from the fact that with Definition 2 of halting, only the cell state, rather than a head/cell state combination, determined when a machine should stop. As head states did not explicitly determine TM halting in this definition, they only affected halting probability in the sense that they increased the possible states of the machine. As described in the context of Definition 1 of halting, increasing possible states of a machine may not increase the number of those states that result in halting.

$s \mid k$	2	3	4	5	6
2	0.43210	0.27745	0.19690	0.15535	0.12590
3	0.39370	0.24090	0.17205	0.13495	0.10885
4	0.36475	0.22085	0.15805	0.12745	0.10285
5	0.34470	0.20650	0.14870	0.12005	0.09760
6	0.33055	0.20255	0.14240	0.10835	0.08915

Table 1. Table of halting probabilities from 2,2 TMs to 6,6 TMs. The *s* column represents the total number of head states, and the *k* row represents the total number of cell states. The values represent the halting probability for each respective set of *s*,*k* TMs.

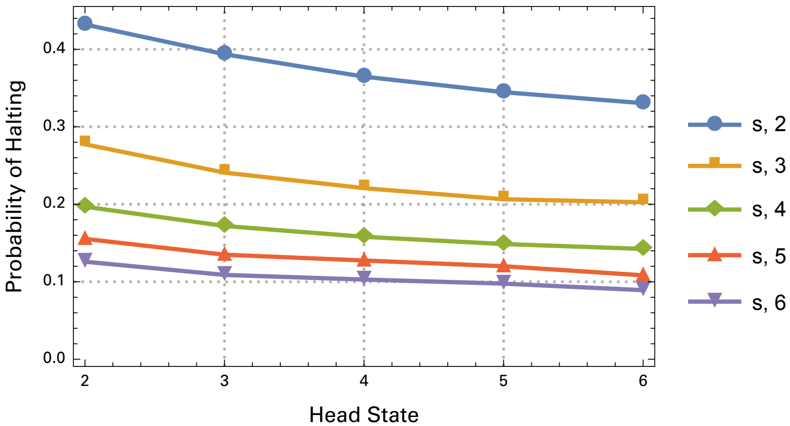


Figure 3. Plot where cell states are held constant while the head states change. The data points have only been joined for visualization purposes. Each line represents a different constant cell state.

Although it is possible to find functions that somewhat resemble the preceding plots, for the most part, polynomial approximations are inaccurate. As these plots refer specifically to the relation the head or cell state has with probability, they are likely irrelevant to the Levin distribution.

https://doi.org/10.25088/ComplexSystems.27.1.85

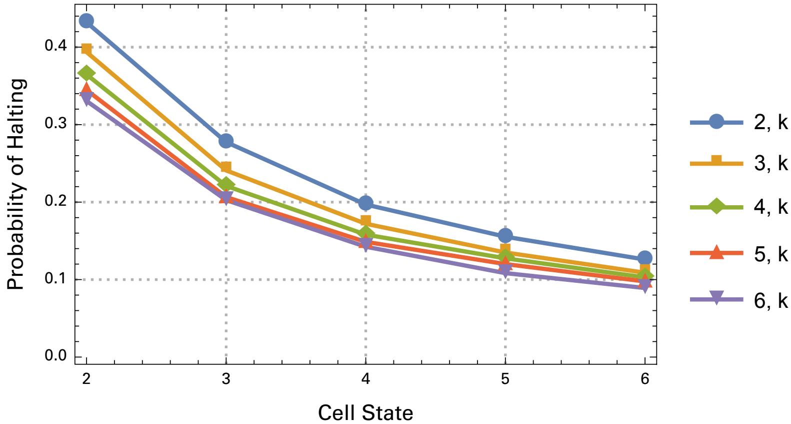


Figure 4. Plot where head states are held constant while the cell states change. The data points have only been joined for visualization purposes. Each line represents a different constant head state.

3.3 Evaluating Halting 2,2 Turing Machines

Figure 5 provides a log plot of the halting times of the 1774 halting 2,2 TMs. The step at which they halt was assessed after testing all 4096 of these TMs.

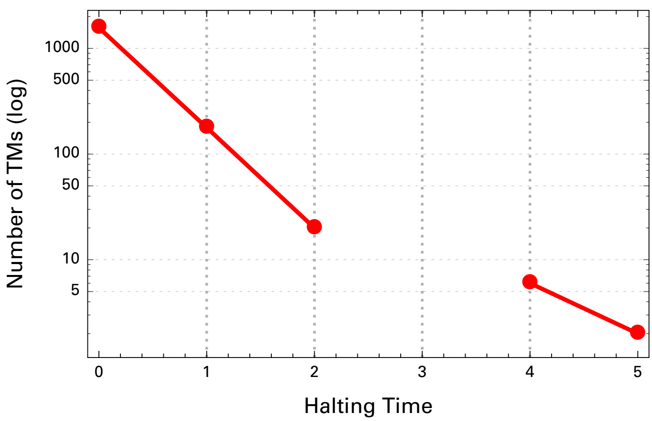
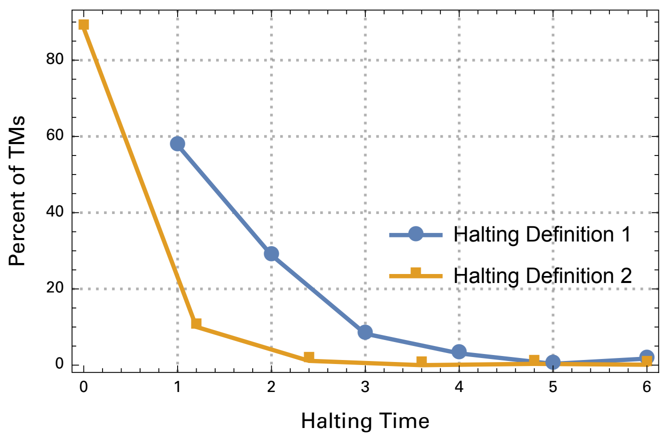


Figure 5. Log plot of the step at which a function halts against how many functions halt at that step. The data points have only been joined for visualization purposes; the gap in the line connecting these points arises from the fact that no TM halted at three steps. This plot fulfills the same function as Figure 2, but represents the values under a different definition of halting.

As mentioned in the context of Figure 2, Figure 5 may approximate the Levin distribution, though due to the halting schematic, it likely varies more than previously mentioned plots.

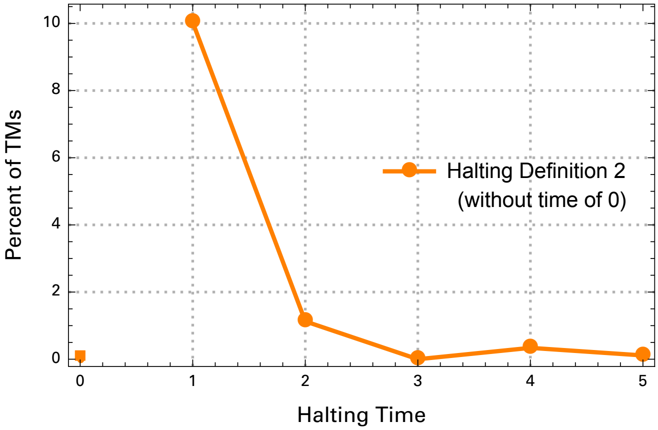
Although the 2,2 halt times from Definition 1 of halting ranged from one to six, the halt times for Definition 2 range from zero to five. This arises mainly through the different definitions of "halting" in each case. For example, it would clearly be impossible for a machine to halt after zero steps using Definition 1; that would imply that the TM halted before it even started. This, however, is acceptable per Definition 2, as this definition describes a change in state rather than a state in itself. That is, when there is no change in state, the machine is said to halt, as opposed to a machine reaching a single "halt state." Furthermore, it appears that 2,2 TMs never halt at three or six steps per Definition 2. This may be a random consequence; the low number of machines that halt at four and five steps would certainly allow it. However, it is interesting how both Definitions 1 and 2 briefly increase after a stretch of decreasing values, though more accurate data is needed for a conclusion; the small number of TMs that halt at later steps increases the inaccuracy of approximations.

Additionally, it can be noticed that although neither definition exceeds six steps to halt, with Definition 2 it is generally more likely that the machine will halt at or before the first step. This may be observed through the low number of TMs halting after step one given by Figure 6.



(a)

Figure 6. (*continues*)



(b)

Figure 6. Line plots representing the distributions of the step at which 2,2 TMs halt. The data points have only been joined for visualization purposes. (a) Represents the halting step distributions using the first and second definitions of halting. (b) Shows the same distribution as the orange plot of (a), but does not include the TMs that halted at step zero, emphasizing proportionately how many TMs halted at step one.

3.4 Evaluating Turing Machines with Greater Numbers of Head and Cell States

Much like the problems faced with gathering data using Definition 1 of halting, it is implausible to completely evaluate more complicated TMs. Although in the preceding example of 2,2 TMs every possible rule set was evaluated, this is clearly impractical for TMs with even slightly more possible head or cell states. Therefore, the approach of random selection was taken.

A list of associations corresponding with all 25 TMs touched upon in Section 3.2, spanning from 2,2 TMs to 6,6 TMs, was created, generating 100 000 TMs per s,k combination and determining the halting times of each (Figure 7).

In all cases, TMs that halt immediately make up a clear majority, often with more than 10 000 TMs. The frequency of TMs that halt immediately is given in Table 2. With head states held constant, the percent of immediately halting TMs decreased as total cell states increased (and likewise for constant cell states), reflecting the pattern noted in Table 1. Furthermore, the range of steps at which TMs halted generally increased according to this same pattern, ranging up to halting times of 98, as given in Table 3.

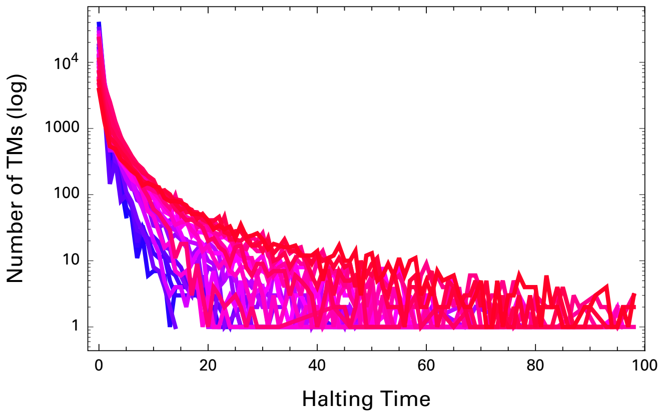


Figure 7. Log plots of the halting times of all 25 measured TMs. The data points have only been joined for visualization purposes. Please note that this figure is only intended to provide a general idea of the spread of halting times. The blue plots represent TMs with fewer possible head/cell state combinations, beginning with 2,2 TMs, 2,3 TMs, and so forth, while the red plots represent TMs with more possible combinations, going down from 6,6 TMs, 5,5 TMs, and so forth.

$s \mid k$	2	3	4	5	6
2	38 067	22 532	15 881	12 298	9928
3	32 198	17 875	11 976	8983	7004
4	28 318	14 666	9656	7107	5497
5	25 342	12 754	8145	5716	4603
6	22 931	11 094	7124	5195	3838

Table 2. Table of the peak values of TM counts for halting steps from 2,2 TMs to 6,6 TMs. The s column represents the total number of head states, and the k row represents the total number of cell states. In all instances, the greatest number of TMs halted at step zero.

Tables 1–3 all demonstrate how increasing possible TM states decreases halting probability. This largely corresponds to observations made previously. In this instance, for a TM to halt, it must reach a particular row of cell states that repeats itself infinitely despite a moving head, essentially creating a loop. The more steps a given loop requires for a TM to reach it, the less likely it is that any random TM will achieve it. This is why immediately halting TMs are far more common: the halting loop is achieved for any TM with {1,0}→{1,0,*off*} in its rule set, where *off* represents any horizontal offset of the head. For a machine to enter a halting loop at a later step, it would have to be the result of all the steps preceding it. The more steps that precede it, the more total results are available at that particular step, leading to a lower halting probability. It follows that increasing the number of possible options available for a TM would likewise compound the number of options for every additional step, decreasing the halting probability further.

$s \mid k$	2	3	4	5	6
2	5	18	30	58	79
3	14	60	74	93	87
4	22	68	94	97	96
5	58	73	95	98	98
6	50	91	92	98	98

Table 3. Table of maximum halting times from 2,2 TMs to 6,6 TMs. The s column represents the total number of head states, and the k row represents the total number of cell states. These TMs were run for only 100 steps, and though some of the greater times, such as 98, suggest that a higher limit may have been more appropriate, none of these values represented more than three TMs out of 100 000, suggesting that selecting larger values would obtain only similarly trivial results.

4. Conclusion

Generally speaking, a Turing machine (TM) will either halt relatively soon or never halt [3]. Per Definition 1 of halting and the TM enumeration schematic used in this study, where a halt state is defined in the TM rule set, 2,2 TMs have a 43.3% chance of halting. A given halting TM is more likely to halt at the first step than any other; that is, the halting probability of a TM decreases with time and will have a smaller chance of halting at every step it progresses, reflecting previous observations such as in [4] or [7]. Furthermore, the distribution of these halting times likely approximated the Levin universal semimeasure distribution. The halting probabilities for 3,2 and 2,3 TMs were, respectively, 40.9% and 32.52%. Computational limitations prevent TMs with four or more options for head or cell state from being analyzed, even through random selection. However, from the limited data collected, it appears that increasing the possible TM states decreases the halting probability, and that possible cell states affect this more than possible head states.

Using Definition 2 of halting, where a TM halts when it reaches a fixed point where the cell state no longer changes, approximate halting probabilities for TMs ranging from 2,2 to 6,6 TMs were found. These values are displayed in Table 1. As either the possible head or cell states increase, the halting probability decreases, reflecting the trend observed with Definition 1. In general, it is far more likely for a

TM to halt at or before a time of one than at any other step, with the probability of a TM halting at a later step decreasing at a noticeably faster rate than under Definition 1. Despite this, TMs were found that did not halt until step 98, which is near the limit placed on runtime. Overall, this reflects the aforementioned observations and demonstrates the fact that although the general halting probability of a TM is undecidable due to the halting problem, halting probabilities may be approximated for specific cases.

References

[1] J. J. Joosten, F. Soler Toscano and H. Zenil, "Program-Size versus Time Complexity: Slowdown and Speed-Up Phenomena in the Micro-cosmos of Small Turing Machines," *International Journal of Unconventional Computing*, **7**(5), 2011 pp. 353–387.

[2] S. Wolfram, *A New Kind of Science*, Champaign, IL: Wolfram Media, Inc., 2002 pp. 78–81.

[3] C. S. Calude and M. A. Stay, "Most Programs Stop Quickly or Never Halt," *Advances in Applied Mathematics,* **40**(3), 2008 pp. 295–308. doi:10.1016/j.aam.2007.01.001.

[4] C. S. Calude and M. Dumitrescu, "A Probabilistic Anytime Algorithm for the Halting Problem," *Computability*, 2017 pp. 1–13. doi:10.3233/COM-170073.

[5] J.-P. Delahaye and H. Zenil, "Numerical Evaluation of the Complexity of Short Strings: A Glance into the Innermost Structure of Randomness," *Applied Mathematics and Computation,* **219**(1), 2012 pp. 63–77. doi:10.1016/j.amc.2011.10.006.

[6] F. Soler-Toscano and H. Zenil, "A Computable Measure of Algorithmic Probability by Finite Approximations with an Application to Integer Sequences," *Complexity*, **2017**, Article ID 7208216, 2017. doi:10.1155/2017/7208216.

[7] H. Zenil, "Computer Runtimes and the Length of Proofs," in *Computation, Physics and Beyond* (M. J. Dinneen, B. Khoussainov and A. Nies, eds.), Berlin, Heidelberg: Springer, 2012 pp. 224–240. doi:10.1007/978-3-642-27654-5_ 17.

[8] H. Zenil, F. Soler-Toscano and J. J. Joosten, "Empirical Encounters with Computational Irreducibility and Unpredictability," *Minds and Machines*, **22**(3), 2012 pp. 149–165. doi:10.1007/s11023-011-9262-y.

[9] E. W. Weisstein. "Busy Beaver" from Wolfram MathWorld—A Wolfram Web Resource. mathworld.wolfram.com/BusyBeaver.html.

[10] P. Michel. "The Busy Beaver Competitions." (Jan 5, 2018) www.logique.jussieu.fr/~michel/bbc.html.