THIS DOCUMENT IS THE ONLINE-ONLY APPENDIX TO:

Model Checking of Recursive Probabilistic Systems

KOUSHA ETESSAMI
University of Edinburgh
and
MIHALIS YANNAKAKIS
Columbia University

ACM Transactions on Computational Logic, Vol. V, No. N, Month 20YY, Pages 1–45.

A. MISSING PROOFS.

This electronic appendix provides all proofs that are missing in the main body of the paper.

LEMMA 8. *For all $D \in \mathcal{F}'$, $\rho^{-1}(D) \in \mathcal{F}$ and $\mathbf{Pr}_{\Omega}(\rho^{-1}(D)) = \mathbf{Pr}_{\Omega'}(D)$.*

PROOF. It suffices, by standard facts about probability measure, to prove the claim for cylinders $C(w') \subseteq \Omega'$, where $w' = w_0, \dots w_k$. We use induction on k. The base case ($k = 0$) follows from Lemma 7. Namely, $C(\epsilon) = \Omega'$, and $\rho^{-1}(\Omega') = \Omega \setminus \rho^{-1}(\star)$. Thus $\mathbf{Pr}_{\Omega}(\rho^{-1}(\Omega')) = 1 - \mathbf{Pr}_{\Omega}(\rho^{-1}(\star)) = 1$.

For the induction step, suppose that the claim holds for the prefix $w' = w_0 w_1 \dots w_k$. Let $D[w'] = \rho^{-1}(C(w'))$. Define the event $J_{i,y} \in \mathcal{F}$ to be $J_{i,y} = \{t \in \Omega \mid \rho(t) = w_0 \dots w_i \dots\ ,\text{and } w_i = y\}$. Note that by definition of conditional probability, $\mathbf{Pr}_{\Omega}(D[w'w_{k+1}]) = \mathbf{Pr}_{\Omega}(D[w'])\,\mathbf{Pr}_{\Omega}(J_{k+1,w_{k+1}} \mid D[w'])$.

We want to show that $\mathbf{Pr}_{\Omega}(D[w'w_{k+1}]) = \mathbf{Pr}_{\Omega'}(C(w'w_{k+1}))$. We distinguish three cases, based on what type of edge (w_k, w_{k+1}) is in H_A, as in the proof of Lemma 7.

Case 1: w_k is not a call port. Thus $(w_k, w_{k+1}) \in E_{H_A}$ is an ordinary edge of the summary graph and corresponds to an edge of the RMC inside some component A_i of A. Consider the trajectories $t \in D[w']$. From the definition of ρ we know that after some prefix, such a trajectory t arrives at a vertex $\langle \beta, w_k \rangle$, and subsequently never reaches an exit of A_i, i.e., it retains β as a prefix of the call stack for the remainder of the trajectory. The conditional probability $\mathbf{Pr}_{\Omega}(J_{k+1,w_{k+1}} \mid D[w'])$, is the probability that the $(k+1)$-st step of $\rho(t)$ is w_{k+1}, given that the prefix of $\rho(t)$ is $w_0 w_1, ... w_k$. Note that this conditional probability is independent of the call stack β, and that this process has the Markov property, so that it is also independent of how we arrive at w_k. The conditional probability $\mathbf{Pr}_{\Omega}(J_{k+1,w_{k+1}} \mid D[w'])$ is the probability that the execution from w_k transitions next to w_{k+1} and never reaches an exit of the component A_i, conditioned on the event that it never reaches an exit of A_i. Let $\mathrm{NE}(u) \in \mathcal{F}$ be the event that, starting at a node $\langle \beta, u\rangle$, we will never reach an exit, i.e., $\beta \in B^+$ will forever remain on the call stack; because of the Markovian property, the probability of this event does not depend on β and is equal to $\mathrm{ne}(u)$. Recall also that the conditional probability of an event E_1 given event E_2 is $\mathbf{Pr}_\Omega(E_1 \mid E_2) = \mathbf{Pr}_\Omega(E_1 \cap E_2)/\,\mathbf{Pr}_\Omega(E_2)$.

Since w_k is not a call port, and using the Markovian property, we have:

$$\begin{aligned}
\mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid D[w']) &= \mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid J_{k,w_k})\\
&= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \mid J_{0,w_k}),\ \text{(now assuming } p_{\mathrm{init}}(\langle \epsilon, w_k\rangle) = 1)\\
&= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \cap \mathrm{NE}(w_k))/\,\mathbf{Pr}_\Omega(\mathrm{NE}(w_k))\\
&= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \cap \mathrm{NE}(w_{k+1}))/\,\mathrm{ne}(w_k)\\
&= \mathbf{Pr}_\Omega(J_{1,w_{k+1}})\,\mathbf{Pr}_\Omega(\mathrm{NE}(w_{k+1}))/\,\mathrm{ne}(w_k)\\
&= p_{w_k,w_{k+1}}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)
\end{aligned}$$

Therefore, $\mathbf{Pr}_\Omega(D[w'w_{k+1}]) = \mathbf{Pr}_\Omega(D[w'])p_{w_k,w_{k+1}}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)$. By the induction hypothesis, and the construction of M'_A, $\mathbf{Pr}_{\Omega'}(C(w'w_{k+1})) =$ $\mathbf{Pr}_{\Omega'}(C(w'))p'_{w_k,w_{k+1}} = \mathbf{Pr}_\Omega(D[w'])p_{w_k,w_{k+1}}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}\,w_k = \mathbf{Pr}_\Omega(D[w'w_{k+1}])$.

Case 2: $w_k = (b, en)$ is a call port, and $w_{k+1} = (b, ex)$ is a return port. In this case, the conditional probability $\mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid D[w'])$ is the probability that an execution of the RMC starting at the call port $w_k = (b, en)$ of box b reaches the return port $w_{k+1} = (b, ex)$ of b given that it does not reach an exit of the component of w_k (and w_{k+1}). From the properties of conditional probabilities, this is equal to the probability that an execution of the RMC starting at the call port w_k reaches the return port w_{k+1} and then after that it does not reach an exit of the component divided by the probability that an execution starting at w_k does not reach an exit. Thus, similar to case 1, we have:

$$\begin{aligned}
\mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid D[w']) &= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \cap \mathrm{NE}(w_{k+1}))/\,\mathrm{ne}(w_k),\ \ \text{(assuming } p_{\mathrm{init}}(\langle \epsilon, w_k\rangle) = 1)\\
&= \mathbf{Pr}_\Omega(J_{1,w_{k+1}})\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)\\
&= q^*_{(en,ex)}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)
\end{aligned}$$

Again, $\mathbf{Pr}_\Omega(D[w'w_{k+1}]) = \mathbf{Pr}_\Omega(D[w'])q^*_{(w_k,w_{k+1})}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)$, and by induction, $\mathbf{Pr}_{\Omega'}(C(w'w_{k+1})) = \mathbf{Pr}_{\Omega'}(C(w'))p'_{w_k,w_{k+1}} =$ $\mathbf{Pr}_\Omega(D[w'])q^*_{(w_k,w_{k+1})}\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k) = \mathbf{Pr}_\Omega(D[w'w_{k+1}])$.

Case 3: $w_k = (b, en)$ is a call port, and $w_{k+1} = en$ is the corresponding entry. By the definition of the summarization map ρ^H and ρ, the next vertex w_{k+1} after w_k in $\rho(t)$ is en iff the call of the box b does not terminate. Thus, the conditional probability $\mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid D[w'])$ is the probability that in an execution of the RMC starting at the call port $w_k = (b, en)$, the call of the box b does not terminate, given that the execution does not reach an exit of the component of w_k. Note that every execution, in which the call of the box b does not terminate, obviously does not reach an exit of the component of w_k. Therefore, the conditional probability is equal to the probability that the call of the box b from its call port (b, en) (i.e. the component $Y(b)$ starting at its entry en) does not terminate divided by the probability that an execution starting at w_k does not reach an exit of the component of w_k. That is, we have:

$$\begin{aligned}\mathbf{Pr}_\Omega(J_{k+1,w_{k+1}} \mid D[w']) &= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \mid J_{0,w_k}) \\ &= \mathbf{Pr}_\Omega(J_{1,w_{k+1}} \cap \mathrm{NE}(w_k))/\,\mathbf{Pr}_\Omega(\mathrm{NE}(w_k)),\ \text{(assuming } p_{\text{init}}(\langle\epsilon, w_k\rangle) = 1) \\ &= \mathbf{Pr}_\Omega(J_{1,w_{k+1}})/\,\mathrm{ne}(w_k),\ \text{(because } \mathrm{NE}(w_k) \subseteq J_{1,w_{k+1}}) \\ &= \mathbf{Pr}_\Omega(\mathrm{NE}(w_{k+1}))/\,\mathrm{ne}(w_k) = \mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)\end{aligned}$$

Again, $\mathbf{Pr}_\Omega(D[w'w_{k+1}]) = \mathbf{Pr}_\Omega(D[w'])\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k)$, and $\mathbf{Pr}_{\Omega'}(C(w'w_{k+1})) = \mathbf{Pr}_{\Omega'}(C(w'))p'_{w_k,w_{k+1}} = \mathbf{Pr}_\Omega(D[w'])\,\mathrm{ne}(w_{k+1})/\,\mathrm{ne}(w_k) = \mathbf{Pr}_\Omega(D[w'w_{k+1}])$. □

THEOREM 17. *The qualitative problem of determining whether a given RMC A satisfies a property specified by a Büchi automaton B with probability $= 1$, (i.e., whether $P_A(L(B)) = 1$)) is EXPTIME-complete. Furthermore, this holds even if the RMC is fixed and each component has one entry and one exit. Moreover, the qualitative "emptiness" problem, namely determining whether $P_A(L(B)) = 0$, is also EXPTIME-complete, again even when the RMC is fixed and each component has one entry and one exit.*

PROOF. The EXPTIME upper bound was established in Theorem 16. So we need to establish EXPTIME-hardness.

We begin by proving the result for determining whether $P_A(L(B)) = 1$ in the case where both A and B are part of the input. The case where A is fixed, and the case for qualitative emptiness, $P_A(L(B)) \overset{?}{=} 0$, are variations on the same proof, and we sketch them at the end.

The reduction is from the acceptance problem for alternating linear space bounded Turing machines. As is well known, $\mathrm{ASPACE}(S(n)) = \cup_{c>0}\mathrm{DTIME}(c^{S(n)})$. There is a fixed linear space bounded alternating Turing machine, T, such that the problem of deciding whether T accepts a given input of length n is EXPTIME-complete. We can assume wlog that T has one tape, and uses space n. The tape contains initially the given input x. Recall that an alternating TM has four types of states: existential, universal, accepting and rejecting. We assume wlog that the TM has two possible moves from each existential and universal state, and it halts when it is in an accepting or rejecting state. Let Γ be the tape alphabet, Q the set of states and $\Delta = \Gamma \cup (Q \times \Gamma)$ the extended tape alphabet. A configuration of the TM is expressed as usual as a string of length n where the ith symbol is $(q, X) \in (Q \times \Gamma)$ (we will usually write qX instead of (q, X)) if the head is on the tape cell i, the state is q and the tape symbol is X, and otherwise the ith symbol is the tape symbol X in cell i. The type of a configuration (existential, universal etc) is determined by the type of the state. A *computation* is a sequence of configurations starting from the initial one, according to the transition rules of the TM. We assume wlog that all computations of the TM halt.

There is a natural game associated with an alternating TM between two players, an existential player E and a universal player U. The positions of the game correspond to the configurations. Player E moves at the existential configurations and player U at the universal ones. Accepting configurations are winning positions for player E, and rejecting configurations are winning for player U. An input x is accepted by the TM iff the existential player E has a winning strategy from the initial configuration corresponding to x.

We will construct a RMC, A, and a BA, B, so that A satisfies B with probability 1 iff x is not accepted by T, i.e. E does not have a winning strategy.

Let us first mention that the only thing that will matter about A, is its "structure", i.e., which edges have non-zero probability. We thus describe these edges without defining the probabilities explicitly: any positive probabilities that sum to 1 will do.

The RMC A has an initial component C_0 and a component $C(q, X)$ for each state $q \in Q$ and tape symbol $X \in \Gamma$. The automaton B has an initial state s_0, a final state f which is the only accepting state, and a state (δ, i) for each $\delta \in \Delta$, and $i = 1, \ldots, n$. The alphabet of B is the vertex set of A.

Let q_0 be the initial state of the TM T, and let $x = x_1 \cdots x_n$ be the input. Component C_0 of A has an edge from its entry to a node u_0, an edge from u_0 to a box that is mapped to $C(q_0, x_1)$ and an edge from the exit of the box to an absorbing node v_0 that has a self-loop.

Component $C(q, X)$, where q is an existential state and $X \in \Gamma$, is structured as follows. Suppose that the two moves of the TM T when it is in state q and reads X are $(p_k, Y_k, D_k), k = 1, 2$, where $p_k \in Q$ is the next state, Y_k is the symbol written over X, and $D_k = L/R$ (left/right) is the direction of the head movement. For each $i = 1, .., n$, $k = 1, 2$, and $Z \in \Gamma$, the component has a set of nodes $u[q, X, i, k, Z]$, $v[q, X, i, k, Z]$, and a set of boxes $b[q, X, i, k, Z]$, each mapped to component $C(p_k, Z)$. The entry of the component $C(q, X)$ has edges to each of the nodes $u[q, X, i, k, Z]$, every node $u[q, X, i, k, Z]$ has an edge to the call port of the corresponding box $b[q, X, i, k, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, i, k, Z]$, and each of these nodes has an edge to the exit of the component.

Component $C(q, X)$, where q is a universal state and $X \in \Gamma$, is structured as follows. Let again the two moves of the TM T for q and X be $(p_k, Y_k, D_k), k = 1, 2$. For each $i = 1, .., n$, $k = 1, 2$, and $Z \in \Gamma$, the component has again a set of nodes $u[q, X, i, k, Z]$, $v[q, X, i, k, Z]$, and a set of boxes $b[q, X, i, k, Z]$ mapped to $C(p_k, Z)$, and has in addition one more node $w[q, X]$. The entry of the component $C(q, X)$ has edges to each of the nodes $u[q, X, i, 1, Z]$, every node $u[q, X, i, 1, Z]$ has an edge to the call port of the corresponding box $b[q, X, i, 1, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, i, 1, Z]$, and each of these has an edge to node $w[q, X]$. Node $w[q, X]$ has edges to all the nodes $u[q, X, i, 2, Z]$, every node $u[q, X, i, 2, Z]$ has an edge to the call port of the corresponding box $b[q, X, i, 2, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, i, 2, Z]$, and each of these has an edge to the exit of the component.

Component $C(q, X)$, where q is a halting (accepting or rejecting) state and $X \in \Gamma$ has an edge from its entry to a node $u[q, X]$ and from $u[q, X]$ to the exit of the component.

The transitions of the automaton B are as follows. The initial state s_0 of B transitions on input u_0 to the set of states $\{(q_0 x_1, 1), (x_2, 2), \ldots, (x_n, n)\}$. There are no other transitions out of s_0. The final state f transitions to itself on every input.

Let q be an existential or universal state and suppose that the two moves of the TM T when it is in state q and reads X are $(p_k, Y_k, D_k), k = 1, 2$. On input $u[q, X, i, k, Z]$, a state (δ, j) of B has exactly one transition, as follows: If $j = i$ and $\delta \neq qX$ then it transitions to f; else, if $j = i$ and $\delta = qX$ then it transitions to state (Y_k, i); else, if $((j = i + 1$ and $D_k = R)$ or $(j = i - 1$ and $D_k = L))$ and $\delta = Z$ then it transitions to $(p_k Z, j)$; else, if $((j = i + 1$ and $D_k = R)$ or $(j = i - 1$ and $D_k = L))$ and $\delta \neq Z$ then it transitions to f; else, it transitions to itself, (δ, j). On input $v[q, X, i, k, Z]$, a state (δ, j) of B has the following transition: If $j = i$ then it transitions to (qX, i); else, if $((j = i + 1$ and $D_k = R)$ or $(j = i - 1$ and $D_k = L))$ then it transitions to (Z, j); else, it transitions to itself, (δ, j). All states have a self-loop on input $w[q, X]$, v_0, as well as for all the vertices that are entries and exits of boxes.

Let q be a halting state of the TM. On input $u[q, X]$, a state (δ, j) of B transitions to itself if $\delta \in \Gamma$ or $(\delta = qX$ and q is accepting), and it transitions to f otherwise.

This concludes the definition of the RMC A and the Büchi automaton B. Note that A has a bounded number of components (independent of the length of the input x), and every component has one entry and one exit. Note also that all the transitions of B are deterministic except for the transition of the initial state s_0 on input u_0.

Consider a path of the RMC, and look at the corresponding set P of states of B at each step. At u_0, the set P contains one state (δ, i) for each $i = 1, \ldots, n$ corresponding to the initial configuration of the TM. From then on, it is easy to check that P always contains *at most* one state (δ, i) for each i, and either these states form a configuration of the TM or P contains f. Once f is included in P, then it will stay there forever and any continuation of the path will be accepted by B.

Let us call a path of the RMC *valid* if the set P at the end (and during the path) does not contain f. Consider the game tree G of the game corresponding to the TM T on the given input x: The nodes of the tree are the configurations reached by the TM in its computation, the root is the initial configuration, the children of each node are the two successor configurations, and the leaves correspond to halting configurations. An existential strategy corresponds to a subtree G_E of G that contains one child of each (reachable) existential configuration (nodes that are not reachable any more from the root are not included in G_E). We consider the two children of each node as being ordered according to the indexing $(k = 1, 2)$ of the two moves of the configuration.

We claim that every valid path of the RMC corresponds to a prefix of the depth-first-search traversal of an existential game tree G_E, where all the leaves in the prefix are accepting; and conversely every such prefix of a DFS traversal corresponds to a valid path. Note that when a valid path is at the entry of an existential component $C(q, X)$, in order for it to continue to be valid it must move to a node $u[q, X, i, k, Z]$ such that i is the current position of the head, q and X must be the current state and symbol at cell i, and Z must be the symbol in the tape cell where the head moves next according to move $k = 1$ or 2 of the TM. That is, there are precisely two valid choices corresponding to the two possible moves of the existential player. The transitions of B are defined so that the states of the new current set P form the next configuration as the path of the RMC moves to the box corresponding to the move of the TM. When the path exits the box, if it is still valid, then the set P is the same as when the path entered the box. After the node $v[q, X, i, k, Z]$, the set P is updated to restore the configuration as it was when the component $C(q, x)$ was called. For a universal component $C(q, X)$ there is only one correct choice if the path is to remain valid. If the path exits the component remaining valid, it means that it never went through a rejecting component, i.e., the corresponding subtree of G_E that was traversed has only accepting leaves.

If x is accepted by the TM T, then the existential player has a winning strategy, hence there is a valid path of the RMC that reaches node v_0 of C_0 and stays there forever. Thus, with positive probability the RMC follows this path which is not accepted by B. On the other hand, if x is not accepted by the TM T, then every path becomes eventually invalid (either because it reaches a rejecting component or because one of its transitions does not correspond to a TM move) and hence is accepted by B; thus the probability of acceptance is 1.

We are done with the proof that checking $P_A(L(B)) = 1$ is EXPTIME-hard. By Theorem 16, the problem is also EXPTIME-complete.

We now sketch how a variation of the same proof shows that probabilistic emptiness ($P_A(L(B)) > 0$?) is also EXPTIME-complete.

For each component except C_0, add a direct path from entry to exit $en \rightarrow r \rightarrow ex$ through a new node r where the first edge has probability $> 1/2$. Every state of the Büchi automaton B, goes to f on these intermediate nodes. (The purpose of these paths is to make sure that every component exits with probability 1 - but these are not valid paths). Remove the self loop of v_0, add new nodes y_0,z_0 to C_0, and edges $v_0 \rightarrow y_0 \rightarrow z_0 \rightarrow u_0$ with probability 1. Also add a new state g to B which is the only accepting state (f is not accepting anymore). On input y_0, all states of B die (have no transition) except for f that goes to g. On z_0, g goes to the initial state s_0.

By the previous proof , (1) if input x is accepted by the TM T, the old RMC had a path p from the initial vertex to v_0 such that the corresponding set of states of the automaton at the end (for all possible runs) did not include f. (2) If x is not accepted by the TM T, then for every trajectory of the old RMC, the automaton has a run that gets to f.

Because of the new paths to the exits that we have added, every component exits with probability 1 (this follows from basic facts about RMCs, see [Etessami and Yannakakis 2009]). Hence, infinitely often (i.o.), the trajectory will go to u_0, traverse a path, come out at v_0, go to y_0,z_0, back to u_0, and again all over. If the state set of the Büchi automaton includes f when the path arrives at v_0, then it will go next to g, then reset to the initial state and start again. Therefore, if x is not accepted by the TM T, this will happen every time, hence g will appear i.o. and the probability of acceptance $P_A(L(B) = 1$.

If x is accepted by the TM T, and in some iteration the RMC follows the path p as above then the automaton will die when the path reaches y_0. Every time the process returns to u_0 and tries again, there is positive probability that it will follow the path p, so eventually this will happen at some point with probability 1. When it happens, the automaton will die and hence will not accept the trajectory. Thus, in this case $P_A(L(B)) = 0$.

Next, we briefly sketch how we actually only need a fixed RMC, whose size does not depend on the size of the input tape of the TM. Here is the modification. Drop the tape cell index i from the u and v nodes of A, and add a self loop to these nodes; that is, the u and v nodes have now the form $u[q, X, k, Z], v[q, X, k, Z]$ for $q \in Q, X, Z \in \Gamma, k = 1, 2$. Basically, the RMC is going to guess what is the correct index i of the cell with the tape head, which will be the number of times it loops at the node u (and v). The Büchi automaton states keep track of how many times the RMC goes around the loop at the current vertex $u[q, X, k, Z]$ or $v[q, X, k, Z]$. In other words, the BA states have now, besides extended tape symbol $\delta \in \Delta$ and cell number $i = 1, \ldots, n$, another counter $j = 0, 1, \ldots, n$ for the number of iterations of the self-loop at the current u or v vertex of the RMC. If the RMC performs the wrong number of iterations at the current vertex (stays too long or leaves too early) then the BA transitions to f and the game is in effect over. In particular if the BA is at state (qX, i, j) and the counter j tries to exceed i without the RMC leaving the current vertex $u[...]$, or if it leaves $u[...]$ before j reaches i, then the the BA goes to f. If the RMC leaves the current vertex $u[...]$ exactly at the correct time, then (qX, i, i) makes the right transition to the appropriate state $(Y, i, 0)$ corresponding to the Turing machine move. For the other states (δ, i, j) of the BA, first if δ has a state and is not qX then go to f right away; otherwise, if the state is (δ, l, i) when the RMC moves out of $u[...]$ and $l \neq i$, the state assumes that the RMC moved at the right time (i.e. tape head is at cell i) and acts accordingly: for example if the head is supposed to move left and new state $= p$, new symbol (in new position)$= Z$, then (δ, l, i) transitions to $(\delta, l, 0)$ if $l \neq i - 1$, to f if $l = i - 1$ but $\delta \neq Z$, and to $(pZ, l, 0)$ otherwise. The moves at $v[...]$ that restore the state are similar. □

Theorem 23. *For a fixed Büchi automaton B, given a bounded RMC, A, and a rational value $p \in [0, 1]$, we can decide whether $P_A(L(B)) \geq p$ in time polynomial in $|A|$.*

Proof. If the Büchi automaton B is fixed, then the deterministic automaton B' has bounded size. Taking the product with a bounded RMC A results in another bounded RMC $A \otimes B'$ (note that the number of entries and exits of A gets multiplied by the number of states of B'). The termination probabilities of a bounded RMC are in general irrational, but, as shown in [Etessami and Yannakakis 2009], we can answer in polynomial time comparison questions concerning them, using a procedure for the existential theory of the reals with a bounded number of variables.

We summarize below the method from [Etessami and Yannakakis 2009]. First the bounded RMC ($A \otimes B'$ in this case) is preprocessed to identify and remove the vertex-exit pairs with 0 probability. Now use variables $x_{(en,ex)}$ only for the set D of entry-exit pairs (en, ex) of the components of $A \otimes B'$ that have nonzero probability; note that there is a bounded number d of such pairs. Let x' be the restriction of the variable vector x of vertex-exit probabilities to these variables $x_{(en,ex)}$ for $(en, ex) \in D$. Then the exit probabilities for all the vertex-exit pairs (u, ex) can be expressed as rational functions of these entry-exit variables. Specifically, for every vertex-exit pair (u, ex) (including the entry-exit pairs) we can construct in polynomial time two polynomials $f_{(u,ex)}(x'), g_{(u,ex)}(x')$ such that $q^*_{(u,ex)} = f_{(u,ex)}(q'^*)/g_{(u,ex)}(q'^*)$, where q'^* is the restriction of the vector q^* to the set D of (nonzero) entry-exit pairs. The polynomials $f_{(u,ex)}(x'), g_{(u,ex)}(x')$ have rational coefficients of polynomial bit size, and have total degree at most n, the number of vertices. As shown in [Etessami and Yannakakis 2009], the vector q'^* is the (unique) minimal nonzero solution to the following set $C(x')$ of constraints: $f_{(en,ex)}(x') = g_{(en,ex)}(x') \cdot x_{(en,ex)}$ and $x_{(en,ex)} > 0$ for all entry-exit pairs $(en, ex) \in D$, and $\sum_{ex} x_{(en,ex)} \leq 1$ for all entries en of each component of the RMC. This solution q'^* of $C(x')$ can be extended to compute the vector q^* for all vertex-exit pairs (u, ex) using the equations $q^*_{(u,ex)} = f_{(u,ex)}(q'^*)/g_{(u,ex)}(q'^*)$. Furthermore the constraint set $C'(x)$ has the property that if we take any other solution r' of $C(x')$ and extend it similarly to all vertex-exit pairs, it results in a vector r that is a fixed point of the original set $x = P(x)$ and hence is $r \geq q^*$. We can therefore determine whether $q^*_{(u,ex)} \leq c$ for some vertex exit pair (u, ex) and rational c by adding to the constraint set $C(x')$ the variable $x_{(u,ex)}$ and constraints $f_{(u,ex)}(x') = g_{(u,ex)}(x') \cdot x_{(u,ex)}$, and $x_{(u,ex)} \leq c$, and invoking an algorithm for the existential theory of the reals with a bounded number of variables. Similarly, we can determine if a vertex u is deficient in polynomial time by adding to $C(x')$ variables $x_{u,ex}$ for all exits $ex \in Ex_i$ of the component of u and adding constraints $f_{(u,ex)}(x') = g_{(u,ex)}(x') \cdot x_{(u,ex)}$ for all $ex \in Ex_i$, and the constraint $\sum_{ex \in Ex_i} x_{(u,ex)} < 1$.

Construct now the Markov chain $M'_{A,B}$, which is the conditioned summary chain of the RMC $A \otimes B'$. We know its set of states, which are the deficient states of the RMC $A \otimes B'$, and its transitions. We do not compute explicitly the values of the transition probabilities, which are irrational numbers, but rather compute them symbolically as rational functions of the vector x' of the entry-exit probabilities of the RMC $A \otimes B'$. Namely, note that the non-exit probability $\mathrm{ne}(u)$ of a vertex u of $A \otimes B'$ is $\mathrm{ne}(u) = 1 - \sum_{ex \in Ex_i} f_{(u,ex)}(x')/g_{(u,ex)}(x')$. The polynomials $f_{(u,ex)}(x'), g_{(u,ex)}(x')$ have total degree n, so $\mathrm{ne}(u)$ is a rational function $f_u(x')/g_u(x')$ where f_u, g_u are polynomials of total degree $\leq dn = O(n)$, also with rational coefficients of polynomial bit-size, and f_u, g_u can be easily constructed in polynomial time. It follows from the definition of the conditioned summary chain $M'_{A,B}$ that the transition probabilities are also rational functions of x' that can be constructed in polynomial time.

We determine the accepting states and accepting edges, and thus the accepting bottom SCCs of the chain $M'_{A,B}$. As in the proof of Theorem 21, we define a revised Markov chain $M''_{A,B}$ by removing all accepting bottom SCCs and replacing them with a new absorbing node v^*; all transitions going to accepting bottom SCCs are directed now to v^*. The desired probability $P_A(L(B))$ is equal to the probability that a trajectory of $M''_{A,B}$ starting at the initial state $u^* = (v_0, \{q_0\})$ hits v^*. If we had the transition probabilities explicitly, we would compute this probability $P_A(L(B))$ by setting up and solving a linear system of equations. By Cramer's rule, $P_A(L(B))$ is equal to the ratio of the determinants of two matrices, $det(F)/det(G)$, whose entries are the transition probabilities, and the constants 0,1. Now the transition probabilities are represented symbolically by rational functions in x', so the probability $P_A(L(B))$ is equal to the ratio $det(F(x'))/det(G(x'))$ of the determinants of two matrices $F(x')$, $G(x')$ whose entries are ratios of polynomials of total degree $O(n)$. Clearing the denominators in the matrix $F(x')$, we can write it as $F(x') = F_1(x')/f_2(x')$ where $f_2(x')$ is the product of all the denominators (a polynomial of total degree $O(n^3)$) and $F_1(x')$ is a matrix whose entries are polynomials of total degree at most $O(n^3)$. Since x' has a fixed number d of variables, each of these polynomials has at most $O(n^{3d})$ terms and can be computed explicitly in polynomial time. We have $det(F(x')) = det(F_1(x'))/(f_2(x))^n$. The numerator $det(F_1(x'))$ is a polynomial $f_1(x')$ of total degree at most $O(n^4)$ and has at most $O(n^{4d})$ terms. As in [Etessami and Yannakakis 2009] we can compute $f_1(x')$ explicitly using interpolation, by substituting a sufficient number of tuples for the variables (e.g., $O(n^4)$ values for each variable) and solving a linear system of equations to compute the coefficients of all the possible $O(n^{4d})$ terms of $f_1(x')$. The denominator $(f_2(x))^n$ is also a polynomial of total degree $O(n^4)$ and can be computed easily. Similarly $det(G(x'))$ can be computed as the ratio $g_1(x')/g_2(x')$ of two polynomials, and hence $P_A(L(B)) = f_1(x')g_2(x')/f_2(x')g_1(x') = f(x')/g(x')$ is expressed as the ratio of two polynomials $f(x'), g(x')$ of total degree $O(n^4)$.

We can then test whether $P_A(L(B)) \geq p$ by invoking a procedure for the existential theory of the reals with bounded number of variables on the set of constraints consisting of the system $C(x')$ for the RMC $A \otimes B'$ defined above, constraints $(f_u(x'))^2 > 0$ for all deficient vertices u of the RMC $A \otimes B'$ (recall $\mathrm{ne}(u) = f_u(x')/g_u(x')$, thus $(f_u(x'))^2 > 0$ iff $\mathrm{ne}(u) \neq 0$), $t \cdot g(x') = f(x')$ where t is a new variable that stands for $P_A(L(B))$, and $t \geq p$. The constraints $C(x')$ and $(f_u(x'))^2 > 0$ for deficient vertices u ensure that there is a unique solution which is q'^*, the vector of entry-exit probabilities of $A \otimes B'$, and the constraints $t \cdot g(x') = f(x')$, $t \geq p$ imply that $P_A(L(B)) \geq p$. □

LEMMA 31. *Suppose that C satisfies the three conditions of Theorem 30. For every probable pair (u, t) with $u \in K$, the following are true for each $i = 1, \ldots, n$.*

(1) *There is a node (u, t') of C such that t and t' agree in the first i coordinates.*

(2) *There is a finite path $\alpha(u, t, i)$ of M''_A starting at u such that the type of almost all trajectories of the RMC from u that do not exit u's component, whose summary image has prefix $\alpha(u, t, i)$, agrees with t in the first i coordinates.*

PROOF. We use induction on i. The basis, $i = 1$ is trivial: φ_1 is a proposition and part (1) is satisfied by any node (u, t') of C with first component u. Note that C has such a node since every path of K is the projection of a path of C (by condition (2) and Lemma 29). As for part (2), we let $\alpha(u, t, 1)$ be the trivial path that consists of node u.

For the induction step, the lemma follows trivially if φ_i is a proposition, or node i of the parse tree of φ is labelled with a Boolean connective, or if it is labelled with $\mathcal{U}$ and the value of t_i is determined uniquely by property (3) of consistency, i.e., $\varphi_i = \varphi_j \mathcal{U} \varphi_l$ and $t_l = 1$ or $t_i = t_j = 0$. In these cases, if we have a probable pair (u, t) and a node (u, t') of C such that t and t' agree in the first $i - 1$ coordinates, then they must agree also in the ith coordinate. Also, we may let $\alpha(u, t, i) = \alpha(u, t, i-1)$. There are two remaining cases.

Case 1: i is labelled with the next operator. Suppose that $\varphi_i = \bigcirc \varphi_j$. Let (u, t) be a probable pair and take any typical trajectory X of the RMC starting at u that does not exit u's component and has type t. Consider the summary image $\rho(X)$ of X, let v be the second node of $\rho(X)$ and s the type of the suffix of X from (this occurrence of) v on. Since $u \in K$, K is a bottom SCC of M'_A, and there is an edge $u \rightarrow v$, it follows that also $v \in K$.

Subcase 1.1. Suppose first that u is not a call port. Then v is simply the second vertex of the trajectory X. Clearly, v is in the same component of the RMC as u, the trajectory does not exit v's component and since it is typical, the pair (v, s) is probable. By the induction hypothesis, there is a node (v, s') of C such that s and s' agree in the first $i-1$ coordinates. By condition (2) of the theorem and Lemma 29, (v, s') has an incoming edge from a node (u, t') of C with first component u. This node (u, t') fulfils the required property 1: the first i coordinates of t' are determined from the first $i-1$ coordinates of s' in the same way that the corresponding coordinates of t are determined from s, and note that $t_i = s_j$ and $t'_i = s'_j$, hence $t_i = t'_i$. For part 2, we let $\alpha(u, t, i)$ be $u \rightarrow v$ followed by $\alpha(v, s, i-1)$.

Subcase 1.2. Suppose that u is a call port $u = (b, en)$. The second node v of $\rho(X)$ is either the entry en of the component of A corresponding to the box b, or it is a return port $v = (b, ex)$ of the box. In the first case, the argument is exactly the same as above; note that the suffix of X from $v = en$ on does not exit v's component and (v, s) is a probable pair. So suppose that $v = (b, ex)$ is a return port of the box b, and let π be the prefix of X from u to v. The type t at u is the type that is backwards implied by the path π and the type s. Again, (v, s) is a probable pair and thus C contains a node (v, s') where s' agrees in the first $i-1$ coordinates with s. The equivalence class of the path π corresponds to one of the parallel summary edges of M''_A, say edge a, from u to v. From Lemma 29 it follows that C contains a corresponding edge $(u, t') \rightarrow (v, s')$, such that t' is the type that is backwards implied from the path π and s'. Since s and s' agree in the first $i-1$ coordinates, the same is true for all the types implied at corresponding nodes of the path π, and thus also at u, the first node of the path, as well as at the second node of the path π. Since t_i and t'_i are equal to the respective coordinates l at the second node, it follows that t and t' agree in the first i coordinates. As for part 2, we let $\alpha(u, t, i)$ be the summary edge a from u to v (corresponding to the path π) followed by the path $\alpha(v, s, i-1)$.

Case 2: Node i is labelled with the Until operator. Suppose that $\varphi_i = \varphi_j \mathcal{U} \varphi_l$, and that $t_j = 1, t_l = 0$ (we took care of the other possibilities for t). Take a typical trajectory X of the RMC starting at u that does not exit u's component and has type t. Let $X = \langle \epsilon, u \rangle x_1 x_2 \ldots$, and let $Y = \rho(X) = u y_1 y_2 ...$ be its summary image. We will distinguish cases according to the value of t_i.

Subcase 2.1: $t_i = 1$. Let m be the smallest index such that the suffix $x_m x_{m+1} \ldots$ of X satisfies φ_l; such an index m exists by the definition of $\mathcal{U}$, and for all $k < m$, the corresponding suffix from x_k on satisfies φ_j. Suppose first that the summary image $Y = \rho(X)$ includes the node corresponding to x_m, i.e. $x_m = \langle \beta, v \rangle$ and all subsequent $x_q, q > m$ include the context β. Let $s = t^m$ be the type of the suffix of X from x_m on. Since the trajectory is typical, (v, s) is a probable pair, and the summary chain contains a path π' from u to v (namely, the summary image of the prefix of X up to x_m). Therefore, v is in the same bottom SCC K as u. By the induction hypothesis, C contains a node (v, s') such that s' agrees with s in the first

$i-1$ coordinates. From Lemma 29, the path π' from u to v in K is the projection of a path in C from some node (u,t') to (v,s'). It follows then that t and t' agree in the first i coordinates (they agree on coordinate i because all nodes (z,q) along the path have $q_j = 1$ and the final node has $s'_l = s_l = 1$). We let the path $\alpha(u,t,i)$ be π' followed by the path $\alpha(v,s,i-1)$.

Suppose that the image trajectory $Y = \rho(X)$ in the summary chain does not include the node corresponding to x_m, i.e. it is shortcut by a summary edge (w,v), where $w = (b,en)$, $v = (b,ex)$ for some box b. That is, for some indices $p < m$, $q > m$, we have $x_p = \langle \beta, w\rangle$, $x_q = \langle \beta, v\rangle$ and all states of the trajectory X between x_p and x_q include the context βb. Let $r = t^p$, $s = t^q$. Again $v \in K$ and the pair (v,s) is probable. By the induction hypothesis, C contains a node (v,s') such that s that agrees with s' in the first $i-1$ coordinates. From Lemma 29, the SCC C contains a path from some node (u,t') to (v,s') with projection the path π' of M''_A from u to v corresponding to the prefix of X up to x_q. If we consider this prefix of X up to x_q, substitute s' for the type at x_q in place of s, and then infer backwards the types t'^k at the preceding states $x_k, k < q$, obviously all the types t'^k will agree in the first $i-1$ coordinates with t^k. This implies in particular that the type at x_m will have the lth coordinate $t'^m_l = 1$. Since the jth coordinate in all the preceding states is 1, it follows that $t'_i = 1$, hence t' agrees with t in the first i coordinates. We let again the path $\alpha(u,t,i)$ be π' followed by the path $\alpha(v,s,i-1)$.

Subcase 2.2: $t_i = 0$. Recall that $t_j = 1, t_l = 0$. We consider two further subcases.

Subcase 2.2.1: There is a typical trajectory $X = \langle \epsilon, u\rangle x_1 x_2 \ldots$, starting at u that does not exit u's component, has type t, and some suffix of X from some state x_m on satisfies $\varphi_j = \varphi_l = 0$. The arguments are very similar to the case $t_i = 1$. Consider the summary image $Y = \rho(X)$. Either it contains the node corresponding to x_m or the node is shortcut by a summary edge. Consider the second case; the first case is similar and simpler. For some indices $p < m$, $q > m$, we have $x_p = \langle \beta, u\rangle$, $x_q = \langle \beta, v\rangle$ and all states of the trajectory X between x_p and x_q include the context βb. Let $r = t^p$, $s = t^q$. Again $v \in K$ and the pair (v,s) is probable, so by the induction hypothesis, there is a node $(v,s') \in C$ such that s' agrees with s in the first $i-1$. There is a path in C from some node (u,t') to (v,s') with projection the path π' of K from u to v that is the summary image of the prefix of X up to x_q. Again we can infer backwards the types and conclude that t,t' agree in the first i coordinates.

Subcase 2.2.2: For every typical trajectory X, starting at u that does not exit u's component and has type t, every suffix of X satisfies $\varphi_j = 1$ or $\varphi_l = 1$. Consider such a typical trajectory $X = \langle \epsilon, u\rangle x_1 x_2 \ldots$. Suppose that there is an index m such that the suffix $x_m\ldots.$ satisfies $\varphi_l = 1$, and let m be the smallest such index. Since $\varphi_l = 0$ for smaller indices $k < m$, it follows that $\varphi_j = 1$ for them, hence from the semantics of the Until operator it follows that trajectory X satisfies φ_i, contradicting the assumption that $t_i = 0$. Therefore, it must be the case that every suffix $x_m\ldots.$ satisfies $\varphi_l = 0$ and hence $\varphi_j = 1$. We will argue that for any $v \in K$, every probable pair (v,s) has $s_l = 0, s_j = 1$, and there is no edge $w \to v$ of K that is the projection of a probable summary edge into (v,s) with label l.

Let (v,s) be a probable pair with $v \in K$ and consider the finite path $\alpha(v,s,i-1)$. Every trajectory of the summary chain M''_A starting at u will contain this path as a subpath with probability 1. In other words, for almost every trajectory X of the RMC that starts at u and does not exit u's component, its summary image $\rho(X)$ will contain this path. Since the type of almost all trajectories whose ρ image has prefix $\alpha(v, s, i-1)$ agrees with s in the first $i-1$ coordinates, and since every suffix of X satisfies $\varphi_l = 0$ and $\varphi_j = 1$, it follows that $s_l = 0$ and $s_j = 1$.

Suppose that there is a probable summary edge $(w, r) \to (v, s)$ whose label includes l, and with projection the edge $a = w \to v$ of K. Let π be a $u - v$ path of the RMC corresponding to the summary edge. We know that $r_l = s_l = 0$ and $r_j = s_j = 1$. Consider the path consisting of the summary edge a followed by the path $\alpha(v, s, i-1)$. Every trajectory of the summary chain M''_A starting at u will contain this path as a subpath with probability 1. Thus, almost every non-exiting trajectory X of the RMC starting at u will have an image $\rho(X)$ that contains this path. Let $X = \langle\epsilon, u\rangle x_1 x_2 \dots$ be such a typical trajectory of type t where x_p gets mapped to w in the summary chain, x_q is mapped to v, and the subpath $\pi = x_p \dots x_q$ is mapped to the summary edge $a = w \to v$. We may assume wlog (it happens a.s.) that the type of the suffix from x_q on agrees with s in the first $i-1$ coordinates. If we infer the types along the path π backwards from x_q, some intermediate state x_m of the path will have $t^m_l = 1$ because the summary edge includes label l, and clearly this label depends only on the first $i-1$ coordinates of s. By our assumption, no suffix of the trajectory satisfies $\varphi_j = \varphi_l = 0$. It follows that the whole trajectory satisfies $\varphi_i = \varphi_l \mathcal{U} \varphi_j$, contradicting our assumption that $t_i = 0$. We conclude that there is no probable summary edge $(w, r) \to (v, s)$ in G with label l where $w, v \in K$.

In the same way that we showed that if one node of a SCC of G is probable then all the nodes are probable, we can argue that the same property is true if we restrict attention to the first $i-1$ coordinates of the types. This implies that for all nodes (v, s') of C we have $s'_l = 0$ and $s'_j = 1$. Also, no summary edge $(w, r') \to (v, s')$ is labelled l. (Since $l \leq i-1$, if there was a $w - u$ path π that yielded such a l-labelled summary edge, then the above argument would still apply by restricting types to the first $i-1$ coordinates). By condition (3) of Theorem 30, it follows that all nodes (v, s') of C have their ith coordinate $s'_i = 0$. So we may let (u, t') be the node of C that agrees with (u, t) in the first $i-1$ coordinates. We may take $\alpha(u, t, i) = \alpha(u, t, i-1)$. □

THEOREM 33. *The qualitative problem of determining whether a given RMC A satisfies a LTL formula φ with probability 1 (i.e., whether $P_A(\varphi) = 1$) is EXPTIME-hard (thus EXPTIME-complete). Furthermore, this holds even if the RMC is fixed and each component has one entry and one exit.*

PROOF. We reduce from the acceptance problem for alternating linear space bounded Turing Machines. As is well known, $\mathrm{ASPACE}(S(n)) = \cup_{c>0}\mathrm{DTIME}(c^{S(n)})$. There is a fixed linear space bounded alternating Turing machine, T, such that the problem of deciding whether T acccepts a given input of length n is EXPTIME-complete. We can assume wlog that T has one tape, and uses space n. The tape initially contains the given input x. Recall that an alternating TM has four types of states: existential, universal, accepting and rejecting. We assume wlog that the TM has two possible moves from each existential and universal state, and it halts when it is in an accepting or rejecting state. Let Γ be the tape alphabet, Q the set of states and $\Delta = \Gamma \cup (Q \times \Gamma)$ the extended tape alphabet. A configuration of the TM is expressed as usual as a string of length n where the ith symbol is $(q, X) \in (Q \times \Gamma)$ (we will usually write qX instead of (q, X)) if the head is on the tape cell i, the state is q and the tape symbol is X, and otherwise the ith symbol is the tape symbol X in cell i. The type of a configuration (existential, universal, etc.) is determined by the type of the state. A *computation* is a sequence of configurations starting from the initial one, according to the transition rules of the TM. We assume wlog that all computations of the TM halt.

There is a natural game associated with an alternating TM between two players, an existential player E and a universal player U. The positions of the game correspond to the configurations. Player E moves at the existential configurations and player U at the universal ones. Accepting configurations are winning positions for player E, and rejecting configurations for player U. An input x is accepted by the TM iff the existential player E has a winning strategy from the initial configuration corresponding to x.

We will construct a RMC, A, and a LTL formula φ so that A satisfies φ with probability 1 iff x is not accepted by T, i.e. E does not have a winning strategy.

Let us first mention that the only thing that will matter about A, is its "structure", i.e., which edges have non-zero probability. We thus describe these edges without defining the probabilities explicitly: any probabilites that sum to 1 will do.

The construction of the RMC A is similar to the construction in the proof of Theorem 17. The RMC A has an initial component C_0 and a component $C(q, X)$ for each state $q \in Q$ and tape symbol $X \in \Gamma$. Let q_0 be the initial state of the TM T, and let $x = x_1 \cdots x_n$ be the input. Component C_0 of A has an edge from its entry to a node u_0, an edge from u_0 to a box b_0 that is mapped to $C(q_0, x_1)$ and an edge from the exit of the box to an absorbing node v_0 that has a self-loop.

Component $C(q, X)$, where q is an existential state and $X \in \Gamma$, is structured as follows. Suppose that the two moves of the TM when it is in state q and reads X are $(p_k, Y_k, D_k), k = 1, 2$, where $p_k \in Q$ is the next state, Y_k is the symbol written over X, and $D_k = L/R$ (left/right) is the direction of the head movement. For each $k = 1, 2$, and $Z \in \Gamma$, the component has a set of nodes $u[q, X, k, Z]$, $v[q, X, k, Z]$, and a set of boxes $b[q, X, k, Z]$, each mapped to component $C(p_k, Z)$. The entry $en[q, X]$ of the component $C(q, X)$ has edges to each of the nodes $u[q, X, k, Z]$, every node $u[q, X, k, Z]$ has an edge to itself and to the call port of the corresponding box $b[q, X, k, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, k, Z]$, and each of these nodes has an edge to itself and to the exit $ex[q, X]$ of the component.

Component $C(q, X)$, where q is a universal state and $X \in \Gamma$, is structured as follows. Let again the two moves of the TM for q and X be $(p_k, Y_k, D_k), k = 1, 2$. For each $k = 1, 2$, and $Z \in \Gamma$, the component has again a set of nodes $u[q, X, k, Z]$, $v[q, X, k, Z]$, and a set of boxes $b[q, X, k, Z]$ mapped to $C(p_k, Z)$, and has in addition one more node $w[q, X]$. The entry of the component $C(q, X)$ has edges to each of the nodes $u[q, X, 1, Z]$, every node $u[q, X, 1, Z]$ has an edge to itself and to the call port of the corresponding box $b[q, X, 1, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, 1, Z]$, and each of these has an edge to itself and to node $w[q, X]$. Node $w[q, X]$ has edges to all the nodes $u[q, X, 2, Z]$, every node $u[q, X, 2, Z]$ has an edge to itself and to the call port of the corresponding box $b[q, X, 2, Z]$, the return port of each such box has an edge to the corresponding node $v[q, X, 2, Z]$, and each of these has an edge to itself and to the exit of the component.

Component $C(q, X)$, where q is a halting (accepting or rejecting) state and $X \in \Gamma$ has an edge from its entry to a node $u[q, X]$, which has an edge to itself and to a node $v[q, X]$, and $v[q, X]$ has an edge to itself and to the exit of the component.

We will construct a LTL formula φ such that player E has a winning strategy σ on input x iff there is a path π_σ of the RMC A from the initial state to v_0 after which the process stays at v_0 forever and the path violates φ. Before describing φ, we will describe how the path π_σ is constructed from the winning strategy σ of E.

Consider the game tree G of the game corresponding to the TM on the given input x: The nodes of the tree are the configurations reached by the TM in its computation, the root is the initial configuration, the children of each node are the two successor configurations, and the leaves correspond to halting configurations. An existential strategy σ corresponds to a subtree G_σ of G that contains one child of each (reachable) existential configuration (nodes that are not reachable any more from the root are not included in G_σ). We consider the two children of each node as being ordered according to the indexing ($k = 1, 2$) of the two moves of the configuration. If σ is a winning strategy of player E then all the leaves of G_σ are accepting configurations.

Perform a depth-first-search traversal $\alpha = (a_1, a_2, ...)$ of the existential game tree G_σ. We map this traversal to the path π_σ of the RMC A as follows. Initially, π_σ starts at the initial entry node moves to u_0, enters the box b_0 and it is thus in state $\langle b_0, en[q_0, x_1]\rangle$.

In the general step l, suppose that the traversal α is at a node a_l (initially, a_1 is the root of the tree), and moves next to a_{l+1} which is either a child of a_l or its parent. Suppose that a_l is an existential node and a_{l+1} is its child, corresponding to the kth move ($k = 1$ or 2) of the existential configuration c_l of the TM corresponding to node a_l. Let q be the state of the TM in configuration c_l, and let i be the index of the cell where the tape head is and let X be the symbol at cell i. Then the path π_σ constructed so far is at this point at a state $\langle \beta, en[q, X]\rangle$ where the context β consists of a sequence of boxes corresponding to the sequence of configurations on the path of the tree G_σ from the root to the current node a_l. The path π_σ of the RMC moves from the entry $en[q, X]$ of component $C(q, X)$ to $u[q, X, k, Z]$ where Z is the symbol in the current configuration c_l of the new cell ($i - 1$ or $i + 1$) to which the head will move next according to the kth move of the TM. The path stays at $u[q, X, k, Z]$ for i steps, and then it moves to the entry of the box $b[q, X, k, Z]$. The new state of the trajectory π_σ becomes $\langle \beta b[q, X, k, Z], en[p_k, Z]\rangle$, ready to simulate the next step of the traversal α.

If the current node a_l of the DFS traversal α corresponds to a universal configuration c_l and the next node a_{l+1} is its first child, then the path π_σ is extended similarly from the entry node of the appropriate component. If a_{l+1} is the second child of a_l, then the path is at this point at a state $\langle \beta, w[q, X]\rangle$, where the context β again consists of a sequence of boxes corresponding to the sequence of configurations on the path of the tree G_σ from the root to the current node a_l, and q, X are respectively the state and tape symbol under the head of the current configuration c_l. From there the path is extended similarly by moving to the appropriate successor $u[q, X, 2, Z]$, staying there for i steps (where i is the index of the cell of the tape head) and entering then the box $b[q, X, 2, Z]$, ready for the next step.

If a_l is a leaf, then the path loops i times at $u[q, X]$ and at $v[q, X]$ and then proceeds to the exit of the current component $C(q, X)$ and returns to the return port of the innermost box $b[q', X', k, X]$ from the context. Note that a_{l+1} is the parent of a_l in the tree, and the corresponding configuration has state q', tape head symbol X' and the head is at some cell j. The path π_σ then proceeds to $v[q', X', k, X]$, stays there for j steps and then moves on to its successor, which is either the node $w[q', X']$ (if c_{l+1} is a universal configuration and a_l was its first child) or the exit node $ex[q', X']$.

In general, if the next step $a_l \rightarrow a_{l+1}$ of the DFS traversal is a backtracking step from a node to its parent, the path π_σ of the RMC is extended in a similar way. At the end, when the traversal α returns to the root of the tree, the path π_σ reaches the return port of the box b_0 of the top component C_0, and from there it goes to v_0.

We will construct a LTL formula ξ which says that the path of the RMC is of the form π_σ described above, corresponding to a DFS traversal α of an accepting existential strategy tree G_σ. We let $\varphi = \neg\xi$. Then $P_A(\varphi) < 1$ iff $P_A(\xi) > 0$ iff there is such a path π_σ iff E has a winning strategy, i.e. iff the TM accepts the input x.

For simplicity, we have one proposition for each node of the RMC. The formula ξ is a conjunction of several subformulas. First, we want the path to reach v_0, and we do not want it to go through a rejecting state. So, let $\xi_1 = (\wedge_{q,X} \neg u[q, X])\mathcal{U} v_0$ where the conjunction ranges over all pairs (q, X) with q a rejecting state and $X \in \Gamma$.

Second, we build a formula ξ_2 that ensures that the tape head starts from cell 1 and moves correctly in each step. The position of the tape head is represented by the number of iterations at a vertex $u[]$ or at a vertex $v[]$. Let u be the disjunction of all the $u[]$ propositions (not u_0), and similarly let v be the disjunction of all the $v[]$ propositions (not v_0). The expression $\psi_i = (\neg u) \wedge (\bigwedge_{j=1}^{i} \bigcirc^j u) \wedge (\neg \bigcirc^{i+1} u)$ says that in the next step the path moves to a $u[]$ node and stays there for exactly i steps. Similarly we can define an expression ψ'_i for v. For the initialization part, the formula $\neg u \mathcal{U} \psi_1$ says that the head starts at cell 1, and is included in ξ_2. The formula $\Box[(\neg u \wedge \bigcirc u) \rightarrow \vee_{i=1}^{n} \psi_i]$ says that the path never loops more than n times at a node $u[]$, and is also included in ξ_2, and similarly we include a corresponding formula for v.

For a state-symbol pair (q, X), suppose that the kth transition ($k = 1, 2$) of the TM from (q, X) is (p_k, Y_k, D_k), where D_k is L or R, say for concreteness that $D_k = L$. Then, if the path is at a vertex $u[q, X, k, Z]$ and stays for i steps there, which means that the head is at cell i, we want to ensure that the next time the path goes to a $u[]$ node, it stays there for $i-1$ steps. Let $\chi(i, L) = \psi_i \wedge \bigcirc^{i+1}(\neg u \mathcal{U} \psi_{i-1}))$. We include in ξ_2 the formula $\Box[(\neg u \wedge \bigcirc \bigvee_Z u[q, X, k, Z]) \rightarrow \bigvee_{i=1}^{n} \chi(i, L)]$. We define an analogous expression $\chi(i, R)$ for right moves and include in ξ_2 an analogous formula for transitions where the head moves right.

Similar formulas are defined and included in ξ_2 for the v nodes to ensure that they restore the correct head position during the backtracking when we return from a recursive call. Note that the tag $[q, X, k, Z]$ of the box (and the subsequent v node) tells us which way the head moved when we entered the box, so that we move it in the opposite direction to restore its position.

If q is a universal state, then we include a formula in ξ_2 to ensure that the number of iterations at a node $v[q, X, 1, Z]$ is the same as the number of iterations at the next node $u[q, X, 2, Z']$. For q a halting state, we have a subformula in ξ_2 that matches the number of iterations at the $u[q, X]$ node with those at the next $v[q, X]$ node. The formula ξ_2 is the conjunction of all these subformulas.

Finally, we have a formula ξ_3 which ensures that in every cell i the tape symbol is initially the input symbol x_i and thereafter it is only changed when the head is at that cell. That is, (1) the first time that the head is at cell i (i.e., when ψ_i holds for the first time, if ever) the tape symbol is x_i; this can be expressed as $\Box[\neg\psi_i\mathcal{U}((\psi_i \wedge \bigcirc u[x_i]) \vee v_0)]$, where $u[x_i]$ is the disjunction of all the $u[]$ propositions with tape symbol $X = x_i$. (2) If a step puts a symbol Y at cell i (this happens either at a $u[]$ node that moves to a new configuration or at a $v[]$ node that restores an old configuration), then the next time that the head moves to cell i (if there is a next time) it finds the same symbol Y there. First construct a formula $put(Y, i)$, which is the disjunction of all $\psi_i \wedge \bigcirc u[q, X, k, Z]$, where the kth transition of (q, X) is (p_k, Y, D_k), and of all $\psi'_i \wedge \bigcirc v[q, Y, k, Z]$, for all q, k, Z. Then we have $\Box[put(Y, i) \rightarrow \bigcirc^{i+1}(\neg\psi_i\mathcal{U}((\psi_i \wedge \bigcirc u[Y]) \vee v_0)]$.

The formula ξ is the conjunction of ξ_1, ξ_2 and ξ_3.

A trajectory of the RMC satisfies ξ iff it has as a prefix the path π_σ for a winning strategy σ of the existential player. □

Lemma 34. *Probabilities $P'(u, t)$ satisfy constraints 2a-2c.*

Proof. (2a) is obvious: Every trajectory that starts at u and does not exit must have some type, and the types t for which (u, t) is not probable (for which we did not include variables) have probability 0.

For (2b), consider the typical trajectories X that start at u and do not exit u's component. Then $Y = \rho(X)$ is a trajectory of M'_A. With probability $p'_{u,v}$ the second vertex is v, the trajectory does not exit the component of v (which is the same as that of u), and the trajectory from v on has type s with probability $P'(v, s)$; the type of X will be t iff there is an edge $(u, t) \rightarrow (v, s)$ in H.

For (2c), consider again the typical trajectories X that start at $u = (b, en)$ and do not exit u's component, and let $Y = \rho(X)$. There are two kinds of such trajectories. The first kind consists of those that never exit the box b, that is, they enter the component corresponding to b at the entry node en and never reach an exit. This happens with probability $p'_{u,en}$. The subsequent trajectory from en does not exit its component, and has type s will probability $P'(en, s)$; the type of the whole trajectory X will be t iff there is an edge $(u, t) \rightarrow (en, s)$ in H. The second kind of trajectories X consists of those that eventually exit the box b at some return port $v = (b, ex)$, (i.e. v is the second node of the image trajectory $Y = \rho(X)$ in M'_A), but then the rest of X from v does not reach the exit of the component of v (which is the same as the component of u). This happens with probability $p'_{u,v}$. The rest of the trajectory from v has type s with probability $P'(v, s)$. Then X has type t

if the $u - v$ path that was followed to exit the box b implies back t from s; this happens with probability $p'_{u,v} f_{u,v,t,s}$. □

LEMMA 35. *The system (2) of linear equations in the variables $z(u,t)$ has a unique solution.*

PROOF. From the summary chain M'_A we form a refined chain M''_A as described in the previous section, where we replace every summary edge $u \to v$ of M'_A by a set of parallel edges, one for each equivalence class of $u - v$ paths, and we distribute the transition probability of the edge $u \to v$ among these parallel edges proportionately to the probability of the paths of the RMC that they represent. Then $p'_{u,v} f_{u,v,t,s}$ is the sum of transition probabilities on the parallel edges of M''_A corresponding to the classes where s at v maps back to t at u.

Let us also introduce parallel edges and edge weights in the graph H: Replace every summary edge $(u,t) \to (v,s)$ of H by a set of parallel edges, one for each equivalence class of $u - v$ paths that imply back t at u from s at v. Let H' be the resulting multigraph. Now every edge a' of H' corresponds to a unique edge a of M''_A; give weight to edge a' equal to the transition probability on edge a of M''_A. The edge weights of H' do not make H' into a Markov chain because weights out of a node may not sum to 1. Note that every path of H' corresponds to (we'll say, *projects onto*) a unique path of M''_A. Furthermore, for every node (v,s) of H' and every edge $a = u \to v$ of M''_A, the graph H' contains a unique corresponding edge a' into (v,s); the head of the edge is a node (u,t) for some t.

The proof of the lemma uses a similar technique to that of Proposition 5.11 in [Courcoubetis and Yannakakis 1995]. Write the system of equations (2b-2c) as $\mathbf{z} = B\mathbf{z}$ where $\mathbf{z}$ is the vector of variables $z(u,t)$ and B is the coefficient matrix of the right-hand side. The rows and columns of B are indexed by the probable pairs, and the entry $B[(u,t),(v,s)]$ is equal to the sum of the weights of the edges $(u,t) \to (v,s)$ of H'. If α is a finite path (sequence of edges) of M''_A or H', then we denote by $w(\alpha)$ the product of the probabilities (or weights) of the edges along the path α and call it the weight of α. Consider the jth power B^j of B. Then $B^j[(u,t),(v,s)]$ is the sum of the weights of the paths α' of length j of H' from (u,t) to (v,s). Every such path projects to a unique path α of M''_A from u to v, and α has the same weight.

A trajectory of the (refined) summary Markov chain M''_A starting at any node u hits with probability 1 eventually a bottom SCC K. Recall from Lemma 31 that if v is any node of K and s any type such that (v,s) is probable, then there is a finite path $\alpha(v,s,n)$ such that any trajectory of M''_A from v with prefix $\alpha(v,s,n)$ has type s with probability 1. A trajectory from u that hits K will eventually with probability 1 contain the path $\alpha(v,s,n)$ as a subpath. If β is finite a path of M''_A from a node u that hits a bottom SCC K and includes a subpath $\alpha(v,s,n)$ for some $v \in K$ and type s such that (v,s) is probable, then we will say that β is *determined*. We assign to such a β a unique type t, which is the type that is backwards implied by the prefix from u to the occurrence of v right before the subpath $\alpha(v,s,n)$ and the type s at v. Clearly, H' contains a path corresponding to β starting at (u,t) (the path goes on to (v,s) and continues from there). Furthermore, H' has no path corresponding to β starting at any other node (u,t') for any other type $t' \neq t$. The reason is that such a path would have to go to a node (v, s') with $s' \neq s$ followed by a path corresponding to $\alpha(v, s, n)$; but then (v, s') cannot be a probable pair, because almost all trajectories of M''_A from v with prefix $\alpha(v, s, n)$ have type s.

Let $d_j(u, t, v)$ be the sum of the weights (probabilities) of the paths β of M''_A of length j from u to v that are determined of type t. Let $d_j(u, t) = \sum_v d_j(u, t, v)$, let $d_j(u) = \sum_t d_j(u, t)$, and let $\epsilon_j(u) = 1 - d_j(u)$. The last quantity $\epsilon_j(u)$ is the probability that a path of M''_A of length j starting at u is not determined. Thus, by the definition and our discussion above, $\epsilon_j(u) \to 0$ as $j \to \infty$.

Consider a path β from u to v of length j that is determined of type t, i.e. β contributes weight $w(\beta)$ to $d_j(u, t, v)$. As we said above, no node (u, t') with $t' \neq t$ has a path corresponding to β. For every node (v, s) of H' there is a path ending at (v, s) that corresponds to β; this path has to start at (u, t). Therefore β contributes weight $w(\beta)$ to $B^j[(u, t), (v, s)]$ for every s, and does not contribute to any $B^j[(u, t'), (v, s)$ with $t' \neq t$. Therefore, for any s we have $d_j(u, t, v) \leq B^j[(u, t), (v, s)]$.

Conversely, consider a path β of M''_A that contributes its weight to $B^j[(u, t), (v, s)]$, i.e. β is the projection of a path in H' of length j from (u, t) to (v, s). If β is determined then its type must be t and its weight is included in $d_j(u, t, v)$. The set of paths of length j that are not determined have total weight $\epsilon_j(u)$. Therefore, $B^j[(u, t), (v, s)] \leq d_j(u, t, v) + \epsilon_j(u)$. Since $\lim_{j\to\infty} \epsilon_j(u) = 0$, it follows that $\lim_{j\to\infty}(B^j[(u, t), (v, s)] - d_j(u, t, v)) = 0$.

Note that if a path β is determined then so are all its extensions and they have the same type t. Therefore, $d_j(u, t)$ is a non-decreasing function of j, and since it is bounded from above by 1, it has a limit $d_\infty(u, t)$. If $\mathbf{z}$ is any solution to the system (2) then for any j it satisfies $\mathbf{z} = B^j\mathbf{z}$. Thus, $z(u, t) = \sum_{(v,s)} B^j[(u, t), (v, s)]z(v, s)$ $= \sum_{(v,s)}(B^j[(u, t), (v, s)] - d_j(u, t, v))z(v, s) + \sum_{(v,s)} d_j(u, t, v)z(v, s)$. As j tends to ∞, the first term tends to 0 and the second term tends to $d_\infty(u, t)$. It follows that $z(u, t) = d_\infty(u, t)$. □