Formal Analysis of Rewriting System Representing RNA Folding

Krishnendu Ghosh[1][a] and Julia Goldman[2][b]

[1]*Department of Computer Science, College of Charleston, SC, U.S.A.*

[2]*Department of Mathematics, Texas Christian University, TX, U.S.A.*

Keywords: RNA Folding, Probabilistic Model Checking, Rewriting System, Stochastic Modeling.

Abstract: Prediction of RNA structure is an important problem in understanding biological processes in living organism. Computational models have been created to study the processes with the aim of unravelling the RNA structure. In this work, a novel formalism for formal analysis of RNA structure prediction is described. A graph rewriting system is formalized to represent structural dynamics of RNA structure under uncertainty. Probabilistic model checking is performed on queries seeking structural properties in RNA. Experiments were conducted to evaluate the computational feasibility of the model.

1 INTRODUCTION

The significance of the role of RNA in biological processes such as gene expression and inhibition is immense (Riddihough, 2016). The structural dynamics of RNA provides insights in the biological processes. RNA secondary structure prediction is critical in understanding the function of RNA. The primary structure of RNA is represented by a sequence of the nucleotides- **A, U, G, C**. The RNA secondary structure is formed with the folding of an RNA strand with formation of hydrogen bonds. RNA pseudoknots are formed from the Watson-Crick base pairing. It is accepted that the secondary RNA structure is predicted based on the minimum free energy for stability.

The problem of predicting RNA secondary structure containing pseudoknots is NP-complete for a large number class of pseudoknots (Lyngsø and Pedersen, 2000). The design of secondary structure using the Watson-Crick is NP-complete in a more realistic model of RNA sequence (Bonnet et al., 2020). The prediction of RNA secondary structure is computational intensive and hence, construction of novel methods are necessitated. Machine learning algorithms have been studied for RNA secondary structure prediction (Zhao et al., 2021). Given the black-box nature of deep learning (Sato et al., 2021), it is not that useful for biologist to understand the complete process of the structural dynamics of RNA. Probabilistic models have been useful in modeling RNA secondary structure when different sources of data, such as homologous RNA sequences, thermodynamic parameters of the energy minimization model, are combined to predict structure (Dowell and Eddy, 2004). The goal of this work is to evaluate how the structural change a RNA will go through during change of free energy. It is not possible to get precise value of free energy through experiments for studying the structural dynamics of RNA. Computational methods have been sought to study changes in the RNA structure. Our work leverages on construction of a formalism that is based on rewriting system under uncertainty. The computational challenge is to relate the structural changes with the minimum free energy. We consider the minimum free energy as the reason for the RNA structural changes. The rewriting rules represents the change in the structure from one structure to another. The contribution of this work is to model RNA structural dynamics with a finite state machine under uncertainty and then, apply temporal logic as a querying mechanism to evaluate RNA structural dynamics.

Model checking is a technique that verifies dynamic properties on a finite state machine representing the system. Correctness of software, network protocol and hardware have been verified using model checking. Model checking represents a system symbolically and not explicitly. The time complexity of model checking is polynomial to the size of the model. Properties or specifications are stated in the form of temporal logic formulas which are precise

[a] https://orcid.org/0000-0002-8471-6537

[b] https://orcid.org/0000-0003-1963-5914

Ghosh, K. and Goldman, J.
Formal Analysis of Rewriting System Representing RNA Folding.
DOI: 10.5220/0011734300003414
In *Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2023) - Volume 3: BIOINFORMATICS*, pages 235-242
ISBN: 978-989-758-631-6; ISSN: 2184-4305

properties posed as a query to the finite state machine representation of the system. Probabilistic model checking is performed which is the properties represented by different computational logics are posed as query to the stochastic structures.

We create a formalism to study RNA structural dynamics by representing RNA structures by graph rewriting, and uncertainty in the dynamics is incorporated by using stochastic models. Stochastic models represent uncertainty in the model when RNA strands transitions from one structure to another with different, random rates. Computational feasibility and properties of the model are evaluated by experimentation using software, PRISM (Kwiatkowska, 2003). To the best of our knowledge, this is the first work that demonstrates application of model checking to RNA structure prediction.

2 RELATED WORK

In this section, we describe the related work on modeling of RNA secondary structure prediction based on discrete structures and inferences using logic-based approaches.

Formal language approaches have been investigated in the modeling of RNA structure. An algebraic language for tree representation of RNA secondary structure was described (Quadrini et al., 2019). The operations defined on the language were concatenation, nesting, and crossing. Concatenation was used for motifs where one structure follows another. Nesting was used to show when a structure had been inserted into the hairpin, and crossing was to show the interaction between structures. The three operations are used to create a unique tree representation for each RNA structure. To ensure that all RNA secondary structures could be expressed, the operators were used to represent pseudoknots as a unique combination of hairpins, the most basic loop structure. A novel method to compare RNA secondary structures using specific representations of secondary structures based algebraic tree has been reported (Quadrini et al., 2020). RNA pseudoknots have been modeled using term rewriting (Fu et al., 2008).

Formal grammars have been proposed (Jonoska et al., 2021) for modeling RNA:DNA interactions and the formation of R-loops (3-stranded nucleic acid hybrid structure). RNA folding was modeled as graph transformation in the presence of free energy (Mamuye et al., 2016). In this model, each RNA configuration was represented as a graph and the evolution of configurations was rule based, represented by graph grammar.

SAT solvers were also applied in RNA secondary structure prediction (Ganesh et al., 2012). The user-provided code included structural constraints (biological properties of the RNA structure) and energy constraints (quantitative requirements). Specifically, the work address correct attribution of a structural state to each nucleic acid within an RNA sequence. Danos et al (Danos et al., 2012) construct pathways using a new graph-based semantics system and a rule-based language for protein-protein interactions called Kappa. Single pushout (SPO) is the technique used for this model. This means that there will be a left-hand side, a right-hand side, and a domain of definition. RNA can be described using an alphabet of the nucleotides, **(A,U,G,C)** and its secondary structure can be described by the ways in which the nucleotides bond with each other. Often, the optimal secondary structure is predicted to be the one with minimum free energy (MFE). In the Watson-Crick model this would be the structure with the most base pairs. The prediction of the RNA structure with MFE is evaluated for models that do not contain pseudoknots. This is called the RNA folding problem. Inclusion of pseudoknots in the problem essentially causes it to be NP-complete (Bonnet et al., 2020). The RNA design problem involves finding a sequence of nucleotide that folds into a given secondary structure. RNA Design Extension is the same, except for the added condition that some indices of the sequence must contain a specified base. A sample of Boltzman distribution to generate suboptimal RNA structures has been reported (Rogers et al., 2017). Algorithmic construction of RNA secondary structures was investigated and the result- designing RNA secondary structures in the Watson-Crick model was proved to be NP hard if the input structure was labeled with bases at some designated position (Bonnet et al., 2020).

There is a body of literature of model checking in systems biology, in particular using stochastic models which has has been an active research area for a decade (Kwiatkowska and Thachuk, 2014). Formal modeling such as model checking has been used as a querying mechanisms on models of biochemical pathways (Heath et al., 2008; Chabrier-Rivier et al., 2004).

3 PRELIMINARIES

In this section, we give the definitions on which the formalism for RNA structure prediction is based. The formalism integrates concepts from multiple topics such as stochastic structures- discrete-time Markov chain, continuous-time Markov chain, probabilistic

model checking and graph rewriting.
The state based definition of the stochastic structures such as discrete time Markov chain (Baier et al., 2008) is:

Definition 1. *(Discrete-Time Markov Chain (DTMC))* a discrete-time Markov chain is a tuple: $\mathcal{M}_m\langle S, S_0, \iota_{init}, P, L\rangle$ where:

1. S is a finite set of states.
2. S_0 is the set of initial states.
3. $P : S \times S \rightarrow [0,1]$, where P represents the probability matrix and $\sum_{s,s' \in S} \mathcal{P}(s,s') = 1$.
4. $\iota_{init} : S \rightarrow [0,1]$ where $\sum_{s \in S} \iota_{init}(s) = 1$ is the initial distribution.
5. $L : S \rightarrow 2^{AP}$, where L is a labeling function and AP the set of atomic propositions.

Definition 2. *(Labeled Continuous-Time Markov Chain (LCTMC))* A labeled Continuous-time Markov Chain (Baier et al., 2008) is a tuple, $\mathcal{K} = \langle S, S_0, R, AP, L\rangle$ where:

1. S is a set of states.
2. $S_0 \subset S$ is the set of initial states.
3. $R : S \times S \rightarrow \mathbb{R}_{\geq 0}$ as the rate matrix.
4. $L : S \leftarrow 2^{AP}$ is a labeling function.

The labeled CTMC described in Definition 2 eliminates the requirement $\mathbf{R}(s,s) = \sum_{s \neq s'} \mathbf{R}(s,s')$, unlike non-state based definition of CTMCs. Self-loops are modeled by $\mathbf{R}(s,s') > 0$.

Definition 3. *(Probabilistic Model checking) Given a probabilistic model,$\mathcal{M}_p$ and formula,ϕ ,model checking is the process of computing the answer to the question of whether $\mathcal{M}_p \models \phi$ holds.*

PCTL syntax includes state formulas ϕ and path formulas ψ. Within the formulas, the next, bounded until, and until operators are allowed (Parker, 2003).

3.1 Probabilistic Computation Tree Logic

We describe the syntax and semantics of probabilistic computation tree logic (PCTL) ((Aziz et al., 1995; Hansson and Jonsson, 1994)).The syntax of PCTL is:

$$\phi ::= true \mid p \mid \phi \wedge \phi \mid \neg\phi \mid \mathcal{P}_{\oplus \mathcal{J}}[\psi]$$
$$\psi ::= X\phi \mid \phi\,\mathcal{U}^{\leq k}\phi \mid \phi\,\mathcal{U}\phi$$

where p is an atomic proposition,$\oplus \in \{\leq, <, \geq, >\}, \mathcal{J} \in [0,1]$ and $k \in \mathbb{N}$. ϕ, ψ are state and path formula respectively. ϕ and ψ are state and path formulas respectively. Each of these formulas are interpreted over a DTMC or an MDP. Each state of DTMC or MDP is labeled from the set of atomic proposition. Specification is represented in the form of a state formula. Path formula ψ are preceded by the probability path operator $\mathcal{P}$. Examples of intervals that are bounds for $\mathcal{P}$ are : $\mathcal{P}_{\leq 0.5}(\psi)$ denotes $\mathcal{P}_{[0,0.5]}(\psi)$. DTMC satisfies $\mathcal{P}_{\oplus \mathcal{J}}$ is the probability of a path from s satisfying ψ is in the bound stated by $\oplus p$. The path forumla,$X\phi$ is true if ϕ is satisfied in the next state. The formula $\phi_1\,\mathcal{U}^{\leq k}\phi_2$ is true if ϕ_2 is satisfied within k time-steps and ϕ_1 is true at that point. Similar is the description of $\phi_1\,\mathcal{U}\phi_2$ where ϕ_2 is true some point in future till then ϕ_1 is true.
The semantics of PCTL over DTMC is given by: Given a DTMC, $\mathcal{M}_p = \langle S_0, S, \mathcal{P}, L\rangle$ and a PCTL formula, the notation $s \models \phi$ represents ϕ is satisfied in s. For a given path, π satisfyinng a PCTL path formula, the notation is $\pi \models \psi$. The semantics of PCTL over $\mathcal{M}_p$ (Parker, 2003):
For a path π :

1. $\pi \models X\phi$ iff $\pi(1) \models \phi$.
2. $\pi \models \phi_1\,\mathcal{U}^{\leq k}\phi_2$ iff $\exists i \leq k.(\pi(i) \models \phi_2 \wedge \pi(j) \models \phi_1, \forall j < i$.
3. $\pi \models \phi_1\,\mathcal{U}\phi_2$ iff $\exists k \geq 0, \pi \models \phi_1\,\mathcal{U}^{\leq k}\phi_2$

For a state, $s \in S$:

1. $s \models true, \forall s \in S$.
2. $s \models a$ iff $a \in L(s)$.
3. $s \models \phi_1 \wedge \phi_2$ iff $s \models \phi_1 \wedge s \models \phi_2$.
4. $s \models \neg\phi$ iff $s \not\models \phi$.
5. $s \models \mathcal{P}_{\oplus \mathcal{J}}[\psi]$ iff $p_s(\psi) \oplus p$.

where $p_s(\psi) = Pr_s(\{\pi \in Path(s) \mid \pi \mid\models \psi\})$ where Pr_s is the set of paths consistes of non-empty sequence of states in the DTMC.

CTMCs can be described by two properties: transient behavior and steady-state behavior. Transient behavior describes the system at a particular moment in time, whereas steady-state behavior describes the system in the long-run.
The temporal logic used to specify properties of CTMCs is called continuous stochastic logic (CSL). In addition to the operators used in PCTL, CSL also uses the time-bounded until operator and the steady-state operator $\mathcal{S}$ (Parker, 2003).

3.2 Continuous Stochastic Logic

Model checking on CTMC is performed by continuous stochastic logic (CSL) (Aziz et al., 1996; Baier et al., 1999). The syntax of CSL (Aziz et al., 1996) The syntax of CSL is

$$\phi ::= true \mid a \mid \phi \wedge \phi \mid \neg\phi \mid \mathcal{P}_{\oplus p}[\psi] \mid \mathcal{S}_{\oplus p}[\phi]$$
$$\psi ::= X\phi \mid \phi\,\mathcal{U}^{<=k}\phi \mid \phi\,\mathcal{U}\phi$$

where a is an atomic proposition,$\oplus \in \{\leq,<,\geq,>\}$, $p \in [0,1]$ and $k \in R_{\geq 0}$. ϕ,ψ are state and path formula respectively. ϕ and ψ are state and path formulas respectively. $\mathcal{P}_{\oplus}p\psi]$represents the probability of ϕ satisfied from a given state satisfies the bound $\oplus p$. The bounded until operator $\phi_1 \mathcal{U}^{\leq}k\phi_2$ is valid if ϕ_2 for a time instant in the interval $[0,k]$ and ϕ_1 is valid at all preceding time instants. The other until operator, $\mathcal{U}$ is not dependent. DTMC or MDP satisfies $\mathcal{P}_{\oplus p}$ is the probability of a path from s satisfying ψ is in the bound stated by $\oplus p$ .The path formula,$X\phi$ is true if ϕ is satisfied in the next state. The formula $\phi_1 \mathcal{U}^{\leq k}\phi_2$ is true if ϕ_2 is satisfied within k time-steps and ϕ_1 is true at that point. Similar is the description of $\phi_1 \mathcal{U}\phi_2$ where ϕ_2 is true some point in future till then ϕ_1 is true.

3.3 RNA Probabilistic Rewriting System

We define the language on RNA graphs and graph rewriting: (G Taentzer and K Ehrig, 2006). The dynamics of RNA structure is modeled using rewriting system. In our formalization, we construct RNA structural graph and then, create a model that leverages on the graph rules under uncertainty. Our model integrates representation of RNA graph and uncertainty in the folding of RNA.

Definition 4. *An RNA-graph is a graph $G_r(V,E,L,L_e)$ where vertices represent bases , and edges represent the bonds between bases such that*

1. *V is the set of vertices.*
2. *E is the set of edges and $e \in E$. $e = \langle v,v' \rangle$ and $v,v' \in V$.*
3. *(No self loop) $\nexists \langle v,u \rangle$ such that $v = u$.*
4. *(Labeling function) $L : V \to Ba$ where Ba is set of bases and Ba=$\{A,G,U,C\}$.*
5. *(Edge Labeling function) $L_e : V \to Bo$ where Bo is the set of bonds.*

In the construction of the graph rewriting system, the set of rules are triggered by a probability (Krause and Giese, 2012). The probabilistic folding model is adapted from probabilistic timed graph (Maximova et al., 2018) where $Dist(Z)$ is the set of probability distribution on the set of rules, Z.

Definition 5. *(Probabilistic folding rule) A probabilistic folding rule, $\kappa = \langle G,Z,\mu \rangle$ where*

1. *G is the RNA graph.*
2. *Z is the set of non-empty finite rules such that $G = L$ for all $z = \langle L \xleftarrow{l} K \xrightarrow{r} R \rangle \in Z$, and $\mu \in Dist(Z)$.*

Here, there are multiple right-hand sides,R for a single G. For the RNA structure model, $Z = \{\text{hairpin, bulge, helix, internal loop}\}$.

Rewrite rules generate graph transformations.

Our model focuses on representing the secondary structure of RNA molecules. Secondary structure refers to the ordered sequence of bases and the bonds that connect them.

3.4 Model

The model based on discrete-time Markov chain,$\mathcal{M}$ is the following- A RNA graph, call it a starting graph, G_0 is transformed $\hat{G}_0$ in next state, when one of the rules, $z \in Z$ triggers. The reading of a transition from a state, s to state s' is a RNA graph G_0 under a rule,z is transformed into $\hat{G}_{0i}$ with probability, p_i such that the sum of p_i is 1. Here, $s,s' \in S$ where S is the set of states in $\mathcal{M}$. For the CTMC variant, there is no requirement of sum of p_is should be one. The rates are the labels on the transition. The reading of a transition from a state, s to state s' is a RNA graph G_0 under a rule, z is transformed into $\hat{G}_{0i}$ with rate, q.
The finite state machine,$\mathcal{K}$ of the RNA structural dynamics is represented as: Given a finite set of RNA structure,$S_{rna} = \{st_1, st_2, \ldots st_n\}$ and set of finite minimum free energy, $FE = \{fe_1, fe_2, \ldots, fe_m\}$ where $fe_i \in \mathbb{R}$ and $n,m,i \in \mathbb{N}$. The states of $\mathcal{K}$ are labeled with a structure, $st, st' \in S_{rna}$ and $fe \in FE$. A transition, $s \to s'$ where s,s' are states in $\mathcal{K}$ implies structure,st is transformed to st' in the presence of fe. Here, the label of s is st and fe. The label of s' is st'.

4 SIMULATION

4.1 Data Preparation

The data for the simulation was from RNAeval webserver (RNA,) in a model of RNA structure(Mamuye et al., 2016), The program, part of the ViennaRNA Web Services(Gruber et al., 2008), allows the user to input any RNA sequence, and then server calculates the energy on a given secondary structure. The thermodynamic description given by RNAeval to calculate the free energy of each structure (Mamuye et al., 2016). The free energy values make it possible to choose the optimal structure for each step in the graph transformation. The comparison of predicted optimal structure to the one predicted by the RNAfold web server were validated (Mamuye et al., 2016).

Table 1: Path from state 0 to state 6.

State	Energy(kcal/mol)	w(i)	w(p)	E(p)	Rate from current to next state
0	0.00	1	1	0	1.80E-4
1	4.80	0.000414653	1.000414653	-0.000110966	3.91E-3
2	2.90	0.009047271	1.009461924	-0.002520743	1.25E-3
3	3.60	0.002905783	1.012367707	-0.00329013	1.89E-6
4	7.60	4.41232E-6	1.012372119	-0.003291296	8.15E-6
5	6.70	1.90043E-5	1.012391124	-0.003296321	3.62E-6
6	7.20	8.44358E-6	1.012399567	-0.003298553	terminal state

Table 2: Path from state 0 to state 10.

State	Energy (kcal/mol)	w(i)	w(p)	E(p)	Rate from current to next state
0	0.00	1	1	0	6.35E-3
7	2.60	0.014720154	1.014720154	-0.003911389	7.64E-4
8	3.90	0.001785947	1.0165061	-0.004382081	1.19E-1
9	0.70	0.321177882	1.337683982	-0.77875129	1.85
10	-2.80	93.9761137	95.31379768	-1.219807747	terminal state

4.2 Computational Feasibility of the Model

The sample strand, CUUACCAUCGGGUUAGAGGAG, used for both the DTMC and CTMC model is taken from literature (Mamuye et al., 2016). The energy values of each structure were calculated using the RNAeval server (Gruber et al., 2008). Figure 1 outlines two possible paths in a finite state machines for the RNA folding. The structural dynamics of RNA strand begins in the unfolded state, s_0. States s_6 and s_{10} have self loop which implies that there is no further folding.

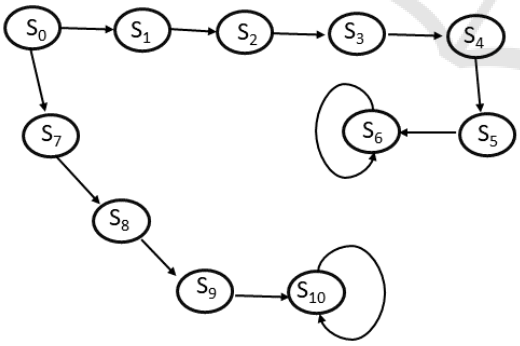

Figure 1: Representation of RNA folding in a finite state machine.

Each arrow represents the formation of a loop and the bonding of bases, and each subsequent state represents the resulting change in the molecule's secondary structure. The two paths in the finite state machines are as follows:

Path from s_0 to s_6:

1. $s_0 \rightarrow s_1$: Bonds form between the 1st and 21st bases and the 2nd and 20th bases to form a helix.
2. $s_1 \rightarrow s_2$: A bond forms between the 3rd and 19th bases to form a helix.
3. $s_2 \rightarrow s_3$: A bond forms between the 6th and 16th bases to form an internal loop.
4. $s_3 \rightarrow s_4$: A bond forms between the 7th and 13th bases to form a bulge.
5. $s_4 \rightarrow s_5$: A bond forms between the 8th and 12th bases to form a hairpin.
6. $s_5 \rightarrow s_6$: A bond forms between the 5th and 18th bases to form a bulge.

Path from s_0 to s_{10}:

1. $s_0 \rightarrow s_7$: Bonds form between the 1st and 16th bases and the 2nd and 15_{th} bases to form a helix.
2. $s_7 \rightarrow s_8$: A bond forms between the 4th and 13th bases to form an internal loop.
3. $s_8 \rightarrow s_9$: A bond forms between the 5th and 12th bases to form a helix.
4. $s_9 \rightarrow s_{10}$: A bond forms between the 6th and 11th bases to form a helix.

Each path is constructed by minimum free energy whose values differ and hence, there are two paths starting from s_0. In the simulation, the structures of RNA are represented symbolically. Note that the sample strand and paths in the finite state machine representing RNA structural is a simple example and the simulations results can be validated by comparing to published values. The sample queries in the form of logic specifications are posed on the stochastic structures representing RNA structural dynamics. The structures are denoted by the states in the queries. The experiments were conducted on system with Intel Core i7 with CPU 2.11 GHz and 16GB RAM. Table 1 and 2 for calculated values of rates and energy where

PCTL Formula	Results	Time (seconds)
P = ? [F x] ”What is the probability of reaching x?	0.027397281 when $x = s_6$ 0.972602719 when $x = s_{10}$	0.001 when $x = s_6$ 0.01 when $x = s_1 0$
P >0.5 [F s=x] ”Verify that the probability of reaching x is greater than 0.5.”	false when $x = s_6$ true when $x = s_{10}$	0.002 when $x = s_6$ 0.009 when $x = s_{10}$
P = ? [s_{10} U s_6] What is the probability that the s_{10} is reached before s_6?	0.027397281	0.006
P = ? [s_6 U s_{10}] ”What is the probability s_6 is reached before state s_{10}?”	0.972602719	0.008
P = ? [s_3 U s_2] ”What is the probability that s_3 before s_2 ?”	0.00	0.005

Figure 2: Execution times and results for PCTL queries on the DTMC model.

CSL Formula	Results	Time (sec)
P = ? [F x_6] meaning: ”What is the probability that the molecule will reach x_6?	0.027397281	0.005
P = ? [F x_{10}] meaning: ”What is the probability that the molecule will reach x_{10}?”	0.972602719	0.002
P = ? [true U[4,4] x_6] meaning: ”What is the probability x_6 exists at time instant 4?”	2.772E-25	0.003
P = ? [true U[4,4] x_{10}] meaning: ”What is the probability of the x_{10} at time instant 4?”	2.948E-6	0.002

Figure 3: Execution times and results for CSL queries on the CTMC model.

$E(p), w(i), w(p)$ denote the energy of the path, weight of the *ith* state and weight of the path, respectively.

4.2.1 DTMC Model

In the DTMC model, each structure Each transition is assigned a probability, defined by the equation given in (Kirkpatrick et al., 2013).

Definition 6. *The equilibrium probability for each state is defined by* $\frac{e^{-E(i)/RT}}{\sum_{j \in S} e^{-E(j)/RT}}$ *where:*

1. *S is the set of states.*
2. $i, j \in S$
3. *$E(i)$ is the energy of state i.*
4. *R is the gas constant. In this case, R is the product of Avogadro's number and the Boltzmann constant.*
5. *T is the temperature. For this model, T is approximately the body temperature, 310.15 K.*

The DTMC model is used to compute the probability of the molecule terminating at either s_6 or s_{10}. The probability of the molecule reaching s_6 is 0.027397281. The probability of the molecule reaching s_{10}, the minimum free energy (MFE) structure, is 0.972602719. The model checker can also indicate whether a structure is likely to occur by verifying whether the probability is greater than one half. This is true only when the final structure is state 10. Additionally, the model can find the probability that one path will terminate before the other, i.e. the probability that s_6 forms before s_{10} and vice versa. An observation that the probability s_6 forms before s_{10} is 0.027397281, and the probability that s_{10} forms before s_6 is 0.972602719. Similarly, queries can confirm the order of states which represents the order of structures formed.. For example, the probability that s_3 forms before s_2 is 0. Figure 2 for execution times for property verification on the model. The sample queries are reachability queries. The time of execution of this simple model is within 1 second.

4.2.2 CTMC Model

In the CTMC model, each structure is represented is labeled to a state. The rates are used from Figure 1 and Figure 2. The CTMC model incorporates transition rates that were calculated by following the process outlined in previous work (Entzian and Raden, 2020): The steps to calculate the transition rates are:

1. The Boltzmann weight, $w(i)$, of each structure is calculated. The weight is defined by $w(i) = e^{-E(i)/RT}$. The terms i, $E(i)$, R, and T are as previously stated in Definition 6.
2. The weight of the path, $w(p)$ is calculated and is the sum of the Boltzmann weights of all structures up to and including that point in the path.
3. The energy of the path is calculated and is defined by $E(p) = -RTlog(w(p))$.

4. The transition rate is defined by a Metropolis rate, represented by $min(1, \frac{E(p)-E(p')}{RT})$.

In addition to probabilities, the CTMC model can be used to incorporate time. For instance, at time instant 4, the probability of the RNA molecule existing in s_6 is $2.772e-25$. Figure 3 shows the times recorded on sample CSL queries on the simulation model. The times for execution for the sample queries is less than 0.01 second. The computational feasibility of the model is efficient for the simple model. Therefore, experiments can be performed on large problem sizes.

5 CONCLUSION

The formalism for RNA structure prediction using graph rewriting provided insights how a computational feasible model can be implemented. The model also demonstrates how uncertainty can be incorporated in the model and can be quantified in terms of the probabilities. A model defined by rewriting rules in the PRISM model checker will become more useful when different initial RNA strands are used as input for validation for the formalism. The PCTL and CSL logics are able to express different but complicated properties of the system. The formalism provides a foundation for a rigorous evaluation of RNA structure prediction. Future work would include experiments on large datasets of RNA structure.

ACKNOWLEDGEMENTS

A part of this project was supported by grant P20GM103499-20 (SC-INBRE) from the National Institute of General Medical Sciences, National Institutes of Health (NIH). Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the NIH. KG was supported by NSF CCF-2227898 for part of the work.

REFERENCES

RNAeval Webserver. http://rna.tbi.univie.ac.at/cgi-bin/RNAWebSuite/RNAeval.cgi , last accessed on 10/17/2022.

Aziz, A., Sanwal, K., Singhal, V., and Brayton, R. (1996). Verifying continuous time markov chains. In *International Conference on Computer Aided Verification*, pages 269–276. Springer.

Aziz, A., Singhal, V., Balarin, F., Brayton, R. K., and Sangiovanni-Vincentelli, A. L. (1995). It usually works: The temporal logic of stochastic systems. In *International Conference on Computer Aided Verification*, pages 155–165. Springer.

Baier, C., Katoen, J.-P., and Hermanns, H. (1999). Approximative symbolic model checking of continuous-time markov chains. In *International Conference on Concurrency Theory*, pages 146–161. Springer.

Baier, C., Katoen, J.-P., and Larsen, K. G. (2008). *Principles of model checking*. MIT press.

Bonnet, E., Rzazewski, P., and Sikora, F. (2020). Designing rna secondary structures is hard. *Journal of Computational Biology*, 27(3):302–316.

Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., and Schächter, V. (2004). Modeling and querying biomolecular interaction networks. *Theoretical Computer Science*, 325(1):25–44.

Danos, V., Feret, J., Fontana, W., Harmer, R., Hayman, J., Krivine, J., Thompson-Walsh, C., and Winskel, G. (2012). Graphs, rewriting and pathway reconstruction for rule-based models. In *FSTTCS 2012-IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 18, pages 276–288.

Dowell, R. D. and Eddy, S. R. (2004). Evaluation of several lightweight stochastic context-free grammars for rna secondary structure prediction. *BMC bioinformatics*, 5(1):1–14.

Entzian, G. and Raden, M. (2020). pourrna—a time-and memory-efficient approach for the guided exploration of rna energy landscapes. *Bioinformatics*, 36(2):462–469.

Fu, X., Wang, H., Harrison, R. W., and Harrison, W. L. (2008). A rule-based approach for rna pseudoknot prediction. *International journal of data mining and bioinformatics*, 2(1):78–93.

G Taentzer, U. P. and K Ehrig, H. E. (2006). Fundamentals of algebraic graph transformation. with 41 figures (monographs in theoretical computer science. an eatcs series).

Ganesh, V., O'donnell, C. W., Soos, M., Devadas, S., Rinard, M. C., and Solar-Lezama, A. (2012). Lynx: A programmatic sat solver for the rna-folding problem. In *International Conference on Theory and Applications of Satisfiability Testing*, pages 143–156. Springer.

Gruber, A. R., Lorenz, R., Bernhart, S. H., Neuböck, R., and Hofacker, I. L. (2008). The vienna rna websuite. *Nucleic acids research*, 36(suppl_2):W70–W74.

Hansson, H. and Jonsson, B. (1994). A logic for reasoning about time and reliability. *Formal aspects of computing*, 6(5):512–535.

Heath, J., Kwiatkowska, M., Norman, G., Parker, D., and Tymchyshyn, O. (2008). Probabilistic model checking of complex biological pathways. *Theoretical Computer Science*, 391(3):239–257.

Jonoska, N., Obatake, N., Poznanović, S., Price, C., Riehl, M., and Vazquez, M. (2021). Modeling rna: Dna hybrids with formal grammars. In *Using Mathematics to Understand Biological Complexity*, pages 35–54. Springer.

Kirkpatrick, B., Hajiaghayi, M., and Condon, A. (2013). A new model for approximating rna folding trajectories and population kinetics. *Computational Science & Discovery*, 6(1):014003.

Krause, C. and Giese, H. (2012). Probabilistic graph transformation systems. In *International Conference on Graph Transformation*, pages 311–325. Springer.

Kwiatkowska, M. (2003). Model checking for probability and time: from theory to practice. In *18th Annual IEEE Symposium of Logic in Computer Science, 2003. Proceedings.*, pages 351–360. IEEE.

Kwiatkowska, M. and Thachuk, C. (2014). Probabilistic model checking for biology. In *Software Systems Safety*, pages 165–189. IOS Press.

Lyngsø, R. B. and Pedersen, C. N. (2000). Pseudoknots in rna secondary structures. In *Proceedings of the fourth annual international conference on Computational molecular biology*, pages 201–209.

Mamuye, A., Merelli, E., and Tesei, L. (2016). A graph grammar for modelling rna folding. *Electronic Proceedings in Theoretical Computer Science*, 231:31–41.

Maximova, M., Giese, H., and Krause, C. (2018). Probabilistic timed graph transformation systems. *Journal of logical and algebraic methods in programming*, 101:110–131.

Parker, D. A. (2003). *Implementation of symbolic model checking for probabilistic systems*. PhD thesis, University of Birmingham.

Quadrini, M., Tesei, L., and Merelli, E. (2019). An algebraic language for rna pseudoknots comparison. *BMC bioinformatics*, 20(4):1–18.

Quadrini, M., Tesei, L., and Merelli, E. (2020). Aspralign: a tool for the alignment of rna secondary structures with arbitrary pseudoknots. *Bioinformatics*, 36(11):3578–3579.

Riddihough, G. (2016). Signals in rna. *Science*, 352(6292):1406–1407.

Rogers, E., Murrugarra, D., and Heitsch, C. (2017). Conditioning and robustness of rna boltzmann sampling under thermodynamic parameter perturbations. *Biophysical journal*, 113(2):321–329.

Sato, K., Akiyama, M., and Sakakibara, Y. (2021). Rna secondary structure prediction using deep learning with thermodynamic integration. *Nature communications*, 12(1):1–9.

Zhao, Q., Zhao, Z., Fan, X., Yuan, Z., Mao, Q., and Yao, Y. (2021). Review of machine learning methods for rna secondary structure prediction. *PLoS computational biology*, 17(8):e1009291.