In silico modelling and analysis of ribosome kinetics and aa-tRNA competition

Bošnački, D.; Pronk, T.E.; de Vink, E.P.

Publication date
2008

Published in
TUCS General Publications

In Silico Modelling and Analysis of Ribosome Kinetics and aa-tRNA Competition

D. Bošnački* [1]
T.E. Pronk† [2]
E.P. de Vink‡ [3]

* Dept. of Biomedical Engineering, Eindhoven University of Technology

† Swammerdam Institute for Life Sciences, University of Amsterdam

‡ Dept. of Mathematics and Computer Science, Eindhoven University of Technology

Abstract

We present a formal analysis of ribosome kinetics using probabilistic model checking and the tool Prism. We compute different parameters of the model, like probabilities of translation errors and average insertion times per codon. The model predicts strong correlation to the quotient of the concentrations of the so-called cognate and near-cognate tRNAs, in accord with experimental findings and other studies. Using piecewise analysis of the model, we are able to give an analytical explanation of this observation.

1 Introduction

The translation mechanism that synthesizes proteins based on mRNA sequences is a fundamental process of the living cell. Conceptually, an mRNA can be seen as a string of codons, each coding for a specific amino acid. The codons of an mRNA are sequentially read by a ribosome, where each codon is translated using an amino acid specific transfer-RNA (aa-tRNA), building one-by-one a chain of amino acids, i.e. a protein. In this setting, aa-tRNA can be interpreted as molecules containing a so-called anticodon, and carrying a particular amino acid. Dependent on the pairing of the codon under translation with the anticodon of the aa-tRNA, plus the stochastic influences such as the changes in the conformation of the ribosome, an aa-tRNA, arriving by Brownian motion, docks into the ribosome and may succeed in adding its amino acid to the chain under construction. Alternatively, the aa-tRNA dissociates in an early or later stage of the translation.

Since the seventies a vast amount of research has been devoted, unraveling the mRNA translation mechanism and related issues. By now, the overall process of translation is reasonably well understood from a qualitative perspective. The translation

[1] Supported by FP6 LTR ESIGNET.
[2] Funded by the BSIK project Virtual Laboratory for e-Science VL-e.
[3] Corresponding author, e-mail evink@win.tue.nl.

process consists of around twenty small steps, a number of them being reversible. For the model organism *Escherichia coli*, the average frequencies of aa-tRNAs per cell have been collected, but regarding kinetics relatively little is known exactly. Over the past few years, Rodnina and collaborators have made good process in capturing the time rates for various steps in the translation process for a small number of specific codons and anticodons [14, 17, 18, 9]. Using various advanced techniques, they were able to show that the binding of codon and anticodon is crucial at a number of places for the time and probability for success of elongation. Based on these results, Viljoen and co-workers started from the assumption that the rates found by Rodnina et al. can be used in general, for all codon-anticodon pairs as estimates for the reaction dynamics. In [7], a complete detailed model is presented for all 64 codons and all 48 aa-tRNA classes for *E. coli*, on which extensive Monte Carlo experiments are conducted. In particular, using the model, codon insertion times and frequencies of erroneous elongations are established. Given the apparently strong correlation of the ratio of so-called near-cognates vs. cognate and pseudo-cognates, and near-cognates vs. cognates, respectively, it is argued that competition of aa-tRNAs, rather than their availability decides both speed and fidelity of codon translation.

In the present paper, we propose to exploit abstraction and model checking of continuous-time Markov chains (CTMCs) with Prism [13, 10]. The abstraction conveniently reduces the number of states and classes of aa-tRNA to consider. The tool provides built-in performance analysis algorithms and path chasing machinery, relieving its user from mathematical calculations. More importantly, from a methodological point of view, the incorporated CSL-logic [2] allows to establish quantitative results for parts of the system, e.g. for first-passage time for a specific state. Such piecewise analysis proves useful when explaining the relationships suggested by the data collected from the model. Additionally, in our case, the Prism tool enjoys rather favourably response times compared to simulation.

Related work The present investigation started from the Monte-Carlo experiments of mRNA translation reported in [7]. A similar stochastic model, but based on ordinary differential equations, was developed in [11]. It treats insertion times, but no translation errors. The model of mRNA translation in [8] assumes insertion rates that are directly proportional to the mRNA concentrations, but assigns the same probability of translation error to all codons.

Currently, there exist various applications of formal methods to biological systems. A selection of recent papers from model checking and process algebra includes [16, 4, 5]. More specifically pertaining to the current paper, [3] applies the Prism modelchecker to analyze stochastic models of signaling pathways. Their methodology is presented as a more efficient alternative to ordinary differential equations models, including properties that are not of probabilistic nature. Also [10] employs Prism on various types of biological pathways, showing how the advanced features of the tool can be exploited to tackle large models.

Organization of the paper Section 2 provides the biological background, discussing the mRNA translation mechanism. Its Prism model is introduced in Section 3. In Section 4, it is explained how error probabilities are obtained from the model and why they correlate with the near-cognate/cognate fraction. This involves adequate estimates of specific stochastic subbehaviour. Insertion times are the subject of Section 5. There too, it is illustrated how the quantitative information of parts of the systems is instrumental in deriving the relationship with the ratio of pseudo-cognate and near-cognates vs. cognates.[4]

Acknowledgments We are grateful to Timo Breit, Christiaan Henkel, Erik Luit,

[4] An appendix presents supplementary figures and data.

Jasen Markovski, and Hendrik Viljoen for fruitful discussions and constructive feedback.

2 A kinetic model of mRNA translation

In nature, there is a fixed correspondence of a codon and an amino acid. This is the well-known genetic code. Thus, an mRNA codes for a unique protein. However, the match of a codon and the anticodon of a tRNA is different from pair to pair. The binding influences the speed of the actual translation.[5] Here, we give a brief overview of the translation mechanism. Our explanation is based on [17, 12]. Two main phases can be distinguished: peptidyl transfer and translocation.

The peptidyl transfer phase runs through the following steps. aa-tRNA arrives at the A-site of the ribosome-mRNA complex by diffusion. The initial binding is relatively weak. Codon recognition comprises (i) establishing contact between the anticodon of the aa-tRNA and the current codon in the ribosome-mRNA complex, and (ii) subsequent conformational changes of the ribosome. *GTP*ase-activation of the elongation factor *EF-Tu* is largely favoured in case of a strong complementary matching of the codon and anticodon. After *GTP*-hydrolysis, producing inorganic phosphate P_i and *GDP*, the affinity of the ribosome for the aa-tRNA reduces. The subsequent accommodation step also depends on the fit of the aa-tRNA.

Next, the translocation phase follows. Another GTP-hydrolysis involving elongation factor *EF-G*, produces *GDP* and P_i and results in unlocking and movement of the aa-tRNA to the P-site of the ribosome. The latter step is preceded or followed by P_i-release. Reconformation of the ribosome and release of *EF-G* moves the tRNA, that has transferred its amino acid to the polypeptide chain, into the E-site of the ribosome. Further rotation eventually leads to dissociation of the used tRNA.

At present, there is little quantitative information regarding the translation mechanism. For *E. coli*, a number of specific rates have been collected [17, 9], whereas some steps are known to be relatively rapid. The fundamental assumption of [7], that we also adopt here, is that experimental data found by Rodnina et al. for the *UUU* and *CUC* codons, extrapolate to other codons as well. However, further assumptions are necessary to fill the overall picture. In particular, Viljoen proposes to estimate the delay due to so-called non-cognate aa-tRNA, that are blocking the ribosomal A-site, as 0.5ms. Also, accurate rates for the translocation phase are largely missing. Again following [7], we have chosen to assign, if necessary, high rates to steps for which data is lacking. This way these steps will not be rate limiting.

3 The Prism model

The abstraction of the biological model as sketched in the previous section is twofold: (i) Instead of dealing with 48 classes of aa-tRNA, that are identified by the their anticodons, we use four types of aa-tRNA distinguished by their matching with the codon under translation. (ii) We combine various detailed steps into one transition. The first reduction greatly simplifies the model, more clearly eliciting the essentials of the underlying process. The second abstraction is more a matter of convenience, though it helps in compactly presenting the model.

For a specific codon, we distinguish four types of aa-tRNA: cognate, pseudo-cognate, near-cognate, non-cognate. Cognate aa-tRNAs have an anticodon that strongly couples with the codon. The amino acid carried by the aa-tRNA is always the right

[5] See Figure 2 and Figure 3 in the appendix.

one, according to the genetic code. The binding of the anticodon of a pseudo-cognate aa-tRNA or a near-cognate aa-tRNA is weaker, but sufficiently strong to occasionally result in the addition of the amino acid to the nascent protein. In case the amino acid of the aa-tRNA is, accidentally, the right one for the codon, we call the aa-tRNA of the pseudo-cognate type. If the amino acid does not coincide with the amino acid the codon codes for, we speak in such a case of a near-cognate aa-tRNA.[6] The match of the codon and the anticodon can be very poor too. We refer to such aa-tRNA as being non-cognate for the codon. This type of aa-tRNA does not initiate a translation step at the ribosome.

The Prism model can be interpreted as the superposition of four stochastic automata, each encoding the interaction of one of the types of aa-tRNA. The automata for the cognates, pseudo-cognates and near-cognates are very similar; the cognate type automaton only differs in its value of the rates from those for pseudo-cognates and near-cognates, while the automata for pseudo-cognates and for near-cognates only differ in their arrival process. The automaton for non-cognates is rather simple.

Below, we are considering average transition times and probabilities for reachability based on exponential distributions. Therefore, following common practice in performance analysis, there is no obstacle to merge two subsequent sequential transitions with rates λ and μ, say, into a combined transition of rate $\lambda\mu/(\lambda+\mu)$. This way, an equivalent but smaller model can be obtained. However, it is noted, that in general, such a simplification is not compositional and should be taken with care.

For the modeling of continuous-time Markov chains, Prism commands have the form `[label] guard` $\rightarrow$ `rate : update ;`. In short, from the commands whose guards are fulfilled in the current state, one command is selected proportional to its relative rate. Subsequently, the update is performed on the state variables. So, a probabilistic choice is made among commands. Executing the selected command results in a progress of time according to the exponential distribution for the particular rate. We refer to [13, 10] for a proper introduction to the Prism modelchecker.

Initially, control resides in the common start state `s=1` of the Prism model with four boolean variables `cogn`, `pseu`, `near` and `nonc` set to false. Next, an arrival process selects one of the booleans that is to be set to true. This is the initial binding of the aa-tRNA. The continuation depends on the type of aa-tRNA: cognate, pseudo-cognate, near-cognate or non-cognate. In fact, a race is run that depends on the concentrations `c_cogn`, `c_pseu`, `c_near` and `c_nonc` of the four types of aa-tRNA and a kinetic constant `k1f`. Following Markovian semantics, the probability in the race for `cogn` to be set to true (the others remaining false) is the relative concentration `c_cogn`/(`c_cogn` + `c_pseu` + `c_near` + `c_nonc`).

```
// initial binding
[ ] (s=1) -> k1f * c_cogn : (s'=2) & (cogn'=true) ;
[ ] (s=1) -> k1f * c_pseu : (s'=2) & (pseu'=true) ;
[ ] (s=1) -> k1f * c_near : (s'=2) & (near'=true) ;
[ ] (s=1) -> k1f * c_nonc : (s'=2) & (nonc'=true) ;
```

As the aa-tRNA, that is just arrived, may dissociate too, the reversed reaction is in the model as well. However, control does not return to the initial state directly, but, for modelchecking purposes, first to the state `s=0` representing dissociation. At the same time, the boolean that was true is reset. Here, cognates, pseudo-cognates and near-cognates are handled with the same rate `k2b`. Non-cognates always dissociate as captured by the separate rate `k2bx`.

```
// dissociation
```

[6] The notion of a pseudo-cognate comes natural in our modeling. However, the distinction between a pseudo-cognate and a near-cognate is non-standard. Usually, a near-cognate refers to both type of tRNA.

```
[ ] (s=2) &  ( cogn | pseu | near ) -> k2b :
        (s'=0) & (cogn'=false) & (pseu'=false) & (near'=false) ;
[ ] (s=2) &  nonc -> k2bx : (s'=0) & (nonc'=false) ;
```

An aa-tRNA that is not a non-cognate can continue from state s=2 in the codon recognition phase, leading to state s=3. This is a reversible step in the translation mechanism, so there are transitions from state s=3 back to state s=2. However, the rates for cognates vs. pseudo- and near-cognates, viz. k3bc, k3bp and k3bn, differ significantly (see Table 1). Note that the values of the booleans do not change.

```
// codon recognition
[ ] (s=2) & ( cogn | pseu | near ) -> k2f : (s'=3) ;
[ ] (s=3) & cogn -> k3bc : (s'=2) ;
[ ] (s=3) & pseu -> k3bp : (s'=2) ;
[ ] (s=3) & near -> k3bn : (s'=2) ;
```

The next forward transition, from state s=3 to state s=4, is a combination of detailed steps involving the processing of GTP. The transition is one-directional, again with a significant difference in the rate k3fc for a cognate aa-tRNA and the rates k3fp and k3fn for pseudo-cognate and near-cognate aa-tRNA, that are equal.

```
// GTPase activation, GTP hydrolysis, EF-Tu conformation change
[ ] (s=3) & cogn -> k3fc : (s'=4) ;
[ ] (s=3) & pseu -> k3fp : (s'=4) ;
[ ] (s=3) & near -> k3fn : (s'=4) ;
```

In state s=4, the aa-tRNA can either be rejected, after which control moves to the state s=5, or accommodates, i.e. the ribosome reconforms such that the aa-tRNA can hand over the amino acid it carries, so-called peptidyl transfer. In the latter case, control moves to state s=6. As before, rates for cognates and those for pseudo-cognates and near-cognates are of different magnitudes.

```
// rejection
[ ] (s=4) & cogn -> k4rc : (s'=5) & (cogn'=false) ;
[ ] (s=4) & pseu -> k4rp : (s'=5) & (pseu'=false) ;
[ ] (s=4) & near -> k4rn : (s'=5) & (near'=false) ;
// accommodation, peptidyl transfer
[ ] (s=4) & cogn -> k4fc : (s'=6) ;
[ ] (s=4) & pseu -> k4fp : (s'=6) ;
[ ] (s=4) & near -> k4fn : (s'=6) ;
```

After a number of movements back-and-forth between state s=6 and state s=7, the binding of the EF-G complex becomes permanent. In the detailed translation mechanism a number of (mainly sequential) steps follows, that are summarized in the Prism model by a single transition to a final state s=8, that represents elongation of the protein in nascent with the amino acid carried by the aa-tRNA. The synthesis is successful if the aa-tRNA was either a cognate or pseudo-cognate for the codon under translation, reflected by either cogn or pseu being true. In case the aa-tRNA was a near-cognate (non-cognates never pass beyond state s=2), an amino acid that does not correspond to the codon in the genetic code has been inserted. In the later case, an insertion error has occurred.

```
// EF-G binding
[ ] (s=6) -> k6f : (s'=7) ;
[ ] (s=7) -> k7b : (s'=6) ;
// GTP hydrolysis, unlocking, tRNA movement and Pi release,
// rearrangements of ribosome and EF-G, dissociation of GDP
[ ] (s=7) -> k7f : (s'=8) ;
```

A number of transitions, linking the dissociation state s=0 and the rejection state s=5 back to the start state s=1, where a race of aa-tRNAs of the four types commences a new, and looping at the final state s=8, complete the Prism model.

```
// no entrance, re-entrance at state 1
[ ] (s=0) -> FAST : (s'=1) ;
// rejection, re-entrance at state 1
[ ] (s=5) -> FAST : (s'=1) ;
// elongation
[ ] (s=8) -> FAST : (s'=8) ;
```

Table 1 collects the rates as gathered from the biological literature [17, 7] and used in the Prism model above.

k1f	140	k3fc	260	k4rc	60	k6f	150
k2f	190	k3fp, k3fn	0.40	k4rp, k4rn	FAST	k7f	145.8
k2b	85	k3bc	0.23	k4fc	166.7	k7b	140
k2bx	2000	k3bp, k3bn	80	k4fp, k4fn	46.1		

Table 1: Rates of the Prism model.

In the next two sections, we will study the Prism model described above for the analysis of the probability for insertion errors, i.e. extension of the peptidyl chain with a different amino acid than the codon codes for, and of the average insertion times, i.e. the average time it takes to process a codon up to elongation.

4 Insertion errors

In this section we show how the model checking features of Prism can be used to predict the misreading frequencies for individual codons. The translation of mRNA into a polypeptide chain is performed by the ribosome machinery with high precision. Experimental measurements show that on average, only one in 10,000 amino acids is added wrongly.[7]

For a codon under translation, a pseudo-cognate anticodon carries precisely the amino acid that the codon codes for. Therefore, successful matching of a pseudo-cognate does not lead to an insertion error. In our model, the main difference of cognates vs. pseudo-cognates and near-cognates is in the kinetics. At various stages of the peptidyl transfer the rates for true cognates differ from the others up to three orders of magnitude.

Figure 1 depicts the relevant abstract automaton, derived from the Prism model discussed above. In case a transition is labeled with two rates, the leftmost number concerns the processing of a cognate aa-tRNA, the rightmost number that of a pseudo-cognate or near-cognate. In three states a probabilistic choice has to be made. The probabilistic choice in state 2 is the same for cognates, pseudo-cognates and near-cognates alike, the ones in state 3 and in state 4 differs for cognates and pseudo-cognates or near-cognates.

For example, after recognition in state 3, a cognate aa-tRNA will go through the hydrolysis phase leading to state 4 for a fraction 0.999 of the cases (computed as $260/(0.23 + 260)$), a fraction being close to 1. In contrast, for a pseudo-cognate or near-cognate aa-tRNA this is 0.005 only. Cognates will accommodate and continue to state 6 with probability 0.736, while pseudo-cognates and near-cognates will do so

[7] Our findings, see Table 4, based on the kinetic rates available are slightly higher.

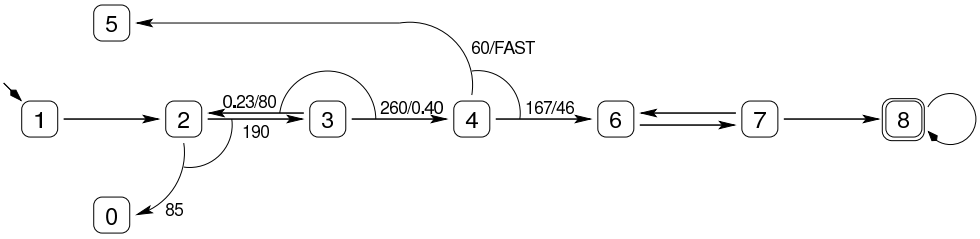

Figure 1: Abstract automaton for error insertion

with the small probability 0.044, the constant FAST being set to 1000 in our experiments. As the transition from state 4 to state 6 is irreversible, the rates of the remaining transitions are not of importance here.

The probability for reaching state 8 in one attempt can be easily computed by Prism via the CSL-formula

```
P=?  [ (s!=0 & s!=5) U (s=8) {(s=2) & cogn} ].
```

The formula asks to establish the probability for all paths where `s` is not set to `0` nor `5`, until `s` have been set to `8`, starting from the (unique) state satisfying `s=2 & cogn`. We obtain $p_s^c = 0.508$, $p_s^p = 0.484 \cdot 10^{-4}$ and $p_s^n = 0.484 \cdot 10^{-4}$, with p_s^c the probability for a cognate to end up in state 8 —and elongate the peptidyl chain— without going through state 0 nor state 5; p_s^p and p_s^n the analogues for pseudo- and near-cognates, respectively. Note that these values are the same for every codon. Different among codons are the concentrations of cognates, pseudo-cognates and near-cognates.[8] Ultimately, the frequencies f_c, f_p and f_n of the types of aa-tRNA in the cell, i.e. the actual number of molecules of the kind, determine the rates for an arrival

As reported in [7], the probability for an erroneous insertion, is strongly correlated with the quotient of the number of near-cognate anticodons and the number of cognate anticodons.[9] In the present setting, this correlation can be formally derived. We have that an insertion error occurs if a near-cognate succeeds to attach its amino acid. Therefore,

$$\begin{aligned} P(\text{error}) &= P(\text{near \& elongation} \mid \text{elongation}) \\ &= \frac{p_s^n \cdot (f_n/tot)}{p_s^c \cdot (f_c/tot) + p_s^p \cdot (f_p/tot) + p_s^n \cdot (f_n/tot)} \approx \frac{p_s^n \cdot f_n}{p_s^c \cdot f_c} \sim \frac{f_n}{f_c} \end{aligned}$$

with $tot = f_c + f_p + f_n$, and where we have used that

$$P(\text{elongation}) = (f_c/tot) \cdot p_s^c + (f_p/tot) \cdot p_s^p + (f_n/tot) \cdot p_s^n$$

and that $p_s^p, p_s^n \ll p_s^c$. Note, the ability to calculate the latter probabilities, illustrating that the approach of piecewise analysis, is instrumental in obtaining the above result.

5 Competition and insertion times

We continue the analysis of the Prism model for translation and discuss the correlation of the average insertion time for the amino acid specified by a codon, on the one hand, and the relative abundance of pseudo-cognate and near-cognate aa-tRNAs, on the other hand. The insertion time of a codon is the average time it takes to elongate the protein in nascent with an amino acid.

The average insertion time can be computed in Prism using the concept of *rewards* (also known as *costs* in Markov theory). Each state is assigned a value as its reward.

[8] See Table 3 in the appendix.

[9] See Figure 4 in the appendix.

Further, the reward of each state is weighted per unit of time. Hence, it is computed by multiplication with the average time spent in the state. The cumulative reward of a path in the chain is defined as a sum over all states in the path of such weighted rewards per state. Thus, by assigning to each state the value 1 as reward, we obtain the total average time for a given path. For example, in Prism the CSL formula `R=? [ F (s=8) ]` which asks to compute the expected time to reach state `s=8`. Recall, in state `s=8` the amino acid is added to the polypeptide chain. So, a script modelchecking the above formula then yields the expected insertion time per codon.[10] A little bit more ingenuity is needed to establish average exit times, for example for a cognate to pass from state `s=2` to state `s=8`. The point is that conditional probabilities are involved. However, since dealing exponential distributions, elimination of transition in favour of adding their rates to that of the remaining ones, does the trick. Various results, some of them used below, are collected in Table 2. (The probabilities of failure and success for the non-cognates are trivial, $p_f^x = 1$ and $p_s^x = 0$, with a time per failed attempt $T_f^x = 0.5 \cdot 10^{-3}$ seconds.)

p_s^c	0.5079	p_f^c	0.4921	T_s^c	0.03182	T_f^c	$9.342 \cdot 10^{-3}$
p_s^p	$4.847 \cdot 10^{-4}$	p_f^p	0.9995	T_s^p	3.251	T_f^p	0.3914
p_s^n	$4.847 \cdot 10^{-4}$	p_f^n	0.9995	T_s^n	3.251	T_f^n	0.3914

Table 2: Exit probabilities and times (in seconds) for three types of aa-tRNA. Failure for exit to states `s=0` or `s=5`; success for exit to state `s=8`.

There is a visible correlation between the quotient of the number of near-cognate aa-tRNA and the number of cognate aa-tRNA.[11] In fact, the average insertion time for a codon is approximately proportional to the near-cognate/cognate ratio. This can be seen as follows. The insertion of the amino acid is completed if state `s=8` is reached, either for a cognate, pseudo-cognate or near-cognate. As we have seen, the probability for the latter two is negligible. Therefore, the number of cognate arrivals is decisive. With p_f^c and p_s^c being the probability for a cognate to fail, i.e. exit at state `s=0` or `s=5`, or to succeed, i.e. reach of state `s=8`, the insertion time T_{ins} can be regarded as a geometric series. (Note the exponent i below.) Important are the numbers of arrivals of the other aa-tRNA types per single cognate arrival, expressed in terms of frequencies. We have

$$\begin{aligned} T_{ins} &= \textstyle\sum_{i=0}^{\top} (p_f^c)^i p_s^c \cdot ((\text{average delay for } i+1 \text{ cognate arrivals}) + T_s^c) \\ &= \textstyle\sum_{i=0}^{\top} (p_f^c)^i p_s^c \cdot (i \cdot (T_f^c + \frac{f_p}{f_c} T_f^p + \frac{f_n}{f_c} T_f^n + \frac{f_x}{f_c} T_f^x) + T_s^c) \\ &\approx \frac{f_p+f_n}{f_c} p_s^c \, T_f^n \textstyle\sum_{i=0}^{\top} i\,(p_f^c)^i \;\sim\; \frac{f_p+f_n}{f_c} . \end{aligned}$$

We have used that T_f^c and T_s^c are negligible, T_f^p equals T_f^n, and $\frac{f_x}{f_c} T_f^x$ is relatively small. Note that the estimate is not accurate for small values of $f_p + f_n$. Nevertheless, closer inspection show that for these values the approximation remains order-preserving. Again, the results obtained for parts of the systems are pivotal in the derivation.

6 Concluding remarks

In this paper, we presented a stochastic model of the translation process based presently available data of ribosome kinetics. We used the CTMC facilities of the Prism tool. Compared to simulation, our approach is computationally more reliable (independent

[10] See Table 5 in the appendix.

[11] See Figure 5 in the appendix.

on the number of simulations) and has faster response times (taking seconds rather then minutes or hours). More importantly, modelchecking allowed us to perform piecewise analysis of the system, yielding better insight in the model compared to just observing the end-to-end results with a monolithic model. Based on this, we improved on earlier observations, regarding error probabilities and insertion times, by actually deriving the correlation suggested by the data. In conclusion, we have experienced aa-tRNA competition as a very interesting biological case study of intrinsic stochastic nature, falling in the category of the well known lambda-phage example [1].

Our model opens a new avenue for future work on biological systems that possess intrinsically probabilistic properties. It would be interesting to apply our method to processes which, similarly to translation, require high precision, like DNA repair, charging of the tRNAs with amino acids, etc. Also, using our model one could check if amino acids with similar biochemical properties substitute erroneously for one another with greater probabilities than dissimilar ones.

References

[1] A. Arkin et al. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected *Escherichia coli* cells. *Genetics*, 149:1633–1648, 1998.

[2] C. Baier et al. Approximate symbolic model checking of continuous-time Markov chains. In *Proc. CONCUR'99*, pages 146–161. LNCS 1664, 1999.

[3] M. Calder et al. Analysis of signalling pathways using continuous time Markov chains. In *Transactions on Computational Systems Biology VI*, pages 44–67. LNBI 4220, 2006.

[4] N. Chabrier and F. Fages. Symbolic model checking of biochemical networks. In *Proc. CMSB 2003*, pages 149–162. LNCS 2602, 2003.

[5] V. Danos et al. Rule-based modelling of cellular signalling. In *Proc. CONCUR*, pages 17–41. LNCS 4703, 2007.

[6] H. Dong et al. Co-variation of tRNA abundance and codon usage in *Escherichia coli* at different growth rates. *Journal of Molecular Biology*, 260:649–663, 1996.

[7] A. Fluitt et al. Ribosome kinetics and aa-tRNA competition determine rate and fidelity of peptide synthesis. *Computational Biology and Chemistry*, 31:335–346, 2007.

[8] M.A. Gilchrist and A. Wagner. A model of protein translation including codon bias, nonsense errors, and ribosome recycling. *Journal of Theoretical Biology*, 239:417–434, 2006.

[9] K.B. Gromadski and M.V. Rodnina. Kinetic determinants of high-fidelity tRNA discrimination on the ribosome. *Molecular Cell*, 13(2):191–200, 2004.

[10] J. Heath et al. Probabilistic model checking of complex biological pathways. In *Proc. CMSB 2006*, pages 32–47. LNBI 4210, 2006.

[11] A.W. Heyd and D.A. Drew. A mathematical model for elongation of a peptide chain. *Bulletin of Mathematical Biology*, 65:1095–1109, 2003.

[12] G. Karp. *Cell and Molecular Biology, 5th ed.* Wiley, 2008.

[13] M. Kwiatkowska et al. Probabilistic symbolic model cheking with Prism: a hybrid approach. *Journal on Software Tools for Technology Transfer*, 6:128–142, 2004. See also `http://www.prismmodelchecker.org/`.

[14] T. Pape et al. Complete kinetic mechanism of elongation factor Tu-dependent binding of aa-tRNA to the A-site of *E. coli*. *EMBO Journal*, 17:7490–7497, 1998.

[15] D. Parker. *Implementation of Symbolic Model Checking for Probabilistic Systems*. PhD thesis, University of Birmingham, 2002.

[16] C. Priami et al. Application of a stochastic name-passing calculus to represent ation and simulation of molecular processes. *Information Processing Letters*, 80:25–31, 2001.

[17] M.V. Rodnina and W. Wintermeyer. Ribosome fidelity: tRNA discrimination, proofreading and induced fit. *TRENDS in Biochemical Sciences*, 26(2):124–130, 2001.

[18] A. Savelsbergh et al. An elongation factor G-induced ribosome rearrangement precedes tRNA–mRNA translocation. *Molecular Cell*, 11:1517–1523, 2003.

Appendix: suplementary figures and data

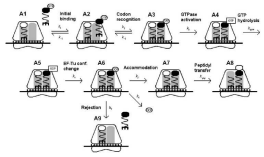

Figure 2: Kinetic scheme of peptidyl transfer taken from [7].

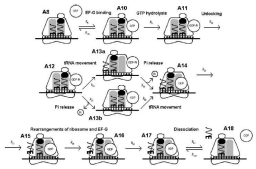

Figure 3: Kinetic scheme of translocation taken from [7].

```
// translation model

stochastic

// constants
const double ONE=1;
const double FAST=1000;

// tRNA rates
const double c_cogn ;
const double c_pseu ;
const double c_near ;
const double c_nonc ;

const double k1f = 140;
const double k2b =  85;
const double k2bx=2000;
const double k2f = 190;
const double k3bc=   0.23;
const double k3bp=  80;
const double k3bn=  80;
const double k3fc= 260;
const double k3fp=   0.40;
const double k3fn=   0.40;
const double k4rc=  60;
const double k4rp=FAST;
const double k4rn=FAST;
const double k4fc= 166.7;
const double k4fp=  46.1;
const double k4fn=  46.1;
const double k6f = 150;
const double k7b = 140;
const double k7f = 145.8;

module ribosome

s : [0..8] init 1 ;
cogn : bool init false ;
pseu : bool init false ;
near : bool init false ;
nonc : bool init false ;

// initial binding
[ ] (s=1) -> k1f * c_cogn : (s'=2) & (cogn'=true) ;
[ ] (s=1) -> k1f * c_pseu : (s'=2) & (pseu'=true) ;
[ ] (s=1) -> k1f * c_near : (s'=2) & (near'=true) ;
[ ] (s=1) -> k1f * c_nonc : (s'=2) & (nonc'=true) ;
[ ] (s=2) &  ( cogn | pseu | near ) -> k2b : (s'=0) &
      (cogn'=false) & (pseu'=false) & (near'=false) ;
[ ] (s=2) &  nonc -> k2bx : (s'=0) & (nonc'=false) ;

// codon recognition
[ ] (s=2) & ( cogn | pseu | near ) -> k2f : (s'=3) ;
[ ] (s=3) & cogn -> k3bc : (s'=2) ;
[ ] (s=3) & pseu -> k3bp : (s'=2) ;
[ ] (s=3) & near -> k3bn : (s'=2) ;

// GTPase activation, GTP hydrolysis, reconformation
[ ] (s=3) & cogn -> k3fc : (s'=4) ;
[ ] (s=3) & pseu -> k3fp : (s'=4) ;
[ ] (s=3) & near -> k3fn : (s'=4) ;

// rejection
[ ] (s=4) & cogn -> k4rc : (s'=5) & (cogn'=false) ;
[ ] (s=4) & pseu -> k4rp : (s'=5) & (pseu'=false) ;
[ ] (s=4) & near -> k4rn : (s'=5) & (near'=false) ;

// accommodation, peptidyl transfer
[ ] (s=4) & cogn -> k4fc : (s'=6) ;
[ ] (s=4) & pseu -> k4fp : (s'=6) ;
[ ] (s=4) & near -> k4fn : (s'=6) ;

// EF-G binding
[ ] (s=6) -> k6f : (s'=7) ;
[ ] (s=7) -> k7b : (s'=6) ;

// GTP hydrolysis, unlocking,
// tRNA movement and Pi release,
// rearrangements of ribosome and EF-G,
// dissociation of GDP
[ ] (s=7) -> k7f : (s'=8) ;

// no entrance, re-entrance at state 1
[ ] (s=0) -> FAST*FAST : (s'=1) ;
// rejection, re-entrance at state 1
[ ] (s=5) -> FAST*FAST : (s'=1) ;
// elongation
[ ] (s=8) -> FAST*FAST : (s'=8) ;

endmodule

rewards
  true : 1;
endrewards
```

codon	cognate	pseudo-cognate	near-cognate	non-cognate	codon	cognate	pseudo-cognate	near-cognate	non-cognate
UUU	1037	0	2944	67493	GUU	5105	0	0	66369
UUC	1037	0	9904	60533	GUC	1265	3840	7372	58997
UUG	2944	0	2324	66206	GUG	3840	1265	1068	65301
UUA	1031	1913	2552	65978	GUA	3840	1265	9036	57333
UCU	2060	344	0	69070	GCU	3250	617	0	67607
UCC	764	1640	4654	64416	GCC	617	3250	8020	59587
UCG	1296	764	2856	66558	GCG	3250	617	1068	66539
UCA	1296	1108	1250	67820	GCA	3250	617	9626	57981
UGU	1587	0	1162	68725	GGU	4359	2137	0	64978
UGC	1587	0	4993	64894	GGC	4359	2137	4278	60700
UGG	943	0	4063	66468	GGG	2137	4359	0	64978
UGA	6219	0	4857	60398	GGA	1069	5427	11807	53171
UAU	2030	0	0	69444	GAU	2396	0	4717	64361
UAC	2030	0	3388	66056	GAC	2396	0	10958	58120
UAG	1200	0	5230	65044	GAG	4717	0	3464	63293
UAA	7200	0	4576	59698	GAA	4717	0	10555	56202
CUU	943	5136	4752	60643	AUU	1737	1737	2632	65368
CUC	943	5136	1359	64036	AUC	1737	1737	6432	61568
CUG	5136	943	2420	62975	AUG	706	1926	4435	64407
CUA	666	5413	1345	64050	AUA	1737	1737	6339	61661
CCU	1301	900	4752	64521	ACU	2115	541	0	68818
CCC	1913	943	2120	66498	ACC	1199	1457	4338	64480
CCG	1481	720	5990	63283	ACG	1457	1199	4789	64029
CCA	581	1620	1430	67843	ACA	916	1740	2791	66027
CGU	4752	639	0	66083	AGU	1408	0	1287	68779
CGC	4752	639	2302	63781	AGC	1408	0	5416	64650
CGG	639	4752	6251	59832	AGG	420	867	6318	63869
CGA	4752	639	2011	64072	AGA	867	420	4248	65939
CAU	639	0	6397	64438	AAU	1193	0	1924	68357
CAC	639	0	3308	67527	AAC	1193	0	6268	64013
CAG	881	764	6648	63181	AAG	1924	0	6523	63027
CAA	764	881	1886	67943	AAA	1924	0	2976	66574

Table 3: Frequencies of cognate, pseudo-cognate, near-cognate and non-cognates for *E. coli* as molecules per cell [6].

UUU	0.002741862683943581	CUU	0.004663729080892617
UUC	0.009117638314789647	CUC	0.0013623408749670932
UUG	7.588473846528858e-4	CUG	4.487561228352708e-4
UUA	0.0023468531911491246	CUA	0.0018888580411442013
UCU	2.8056841829690867e-10	CCU	0.0034116470820387637
UCC	0.005606123319450197	CCC	0.0010419283146932763
UCG	0.002032726835647694	CCG	0.003761852345052361
UCA	9.090727755350428e-4	CCA	0.0022775137744062385
UGU	6.966884002285479e-4	CGU	1.207693755014732e-10
UGC	0.0030362362683066077	CGC	4.587111916100053e-4
UGG	0.003978308597370318	CGG	0.008874544692533565
UGA	7.498426342500918e-4	CGA	3.9837866155798695e-4
UAU	2.8061598550623636e-10	CAU	0.009105588393934699
UAC	0.001568960520388667	CAC	0.004745578685847523
UAG	0.004132405628997547	CAG	0.0069400807775903016
UAA	6.039804446811093e-4	CAA	0.0022666704102712373

GUU	1.122602539973544e-10	AUU	0.0014440395784868422
GUC	0.005495266825145313	AUC	0.0035043308185745276
GUG	2.6820764780942726e-4	AUG	0.005831774423967932
GUA	0.0022306329982350647	AUA	0.0034390541040541776
GCU	1.766661283697676e-10	ACU	2.725325694334536e-10
GCC	0.01245896879253996	ACC	0.0034184472357413403
GCG	3.1789705950373547e-4	ACG	0.003167334470509804
GCA	0.002818616263545499	ACA	0.0029111153328695892
GGU	1.3246548978903072e-10	AGU	8.70279113272123e-4
GGC	9.396128218189778e-4	AGC	0.003719031341166648
GGG	2.7206107910251926e-10	AGG	0.01406993213919797
GGA	0.010230631644252862	AGA	0.004811394879822719
GAU	0.0018570532571304608	AAU	0.0015239834703624298
GAC	0.004322322632194155	AAC	0.00493586499554021
GAG	7.090294740031601e-4	AAG	0.003209595977078994
GAA	0.002136227458736717	AAA	0.0014587873027927622

Table 4: Probabilities per codon for erroneous elongation

UUU	0.3327	CUU	0.8901	GUU	0.0527	AUU	0.2733
UUC	0.8404	CUC	0.6286	GUC	0.7670	AUC	0.4373
UUG	0.1245	CUG	0.1028	GUG	0.1041	AUG	0.8115
UUA	0.4436	CUA	0.9217	GUA	0.2604	AUA	0.4321
UCU	0.0893	CCU	0.4202	GCU	0.0756	ACU	0.0943
UCC	0.7409	CCC	0.1992	GCC	1.5622	ACC	0.4658
UCG	0.3035	CCG	0.4257	GCG	0.1010	ACG	0.4073
UCA	0.2313	CCA	0.5535	GCA	0.3002	ACA	0.5025
UGU	0.1432	CGU	0.0645	GGU	0.0924	AGU	0.1636
UGC	0.3296	CGC	0.1010	GGC	0.1673	AGC	0.3905
UGG	0.4360	CGG	1.3993	GGG	0.2308	AGG	1.4924
UGA	0.1098	CGA	0.0962	GGA	1.2989	AGA	0.5517
UAU	0.0758	CAU	0.8811	GAU	0.2180	AAU	0.2242
UAC	0.2008	CAC	0.5341	GAC	0.4144	AAC	0.4959
UAG	0.4319	CAG	0.7425	GAG	0.1106	AAG	0.3339
UAA	0.0963	CAA	0.4058	GAA	0.2243	AAA	0.1945

Table 5: Estimated average insertion time per codon in seconds

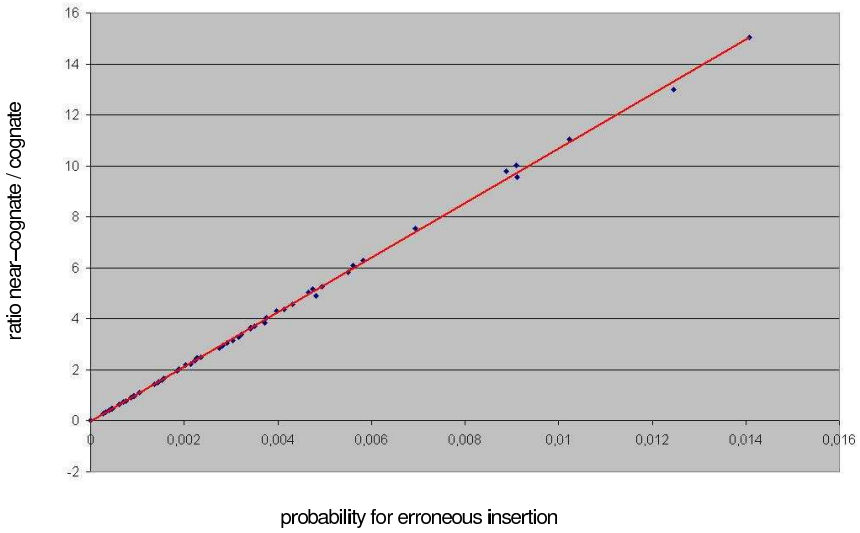

Figure 4: Correlation of $\frac{f_n}{f_c}$ ratio and error probabilities

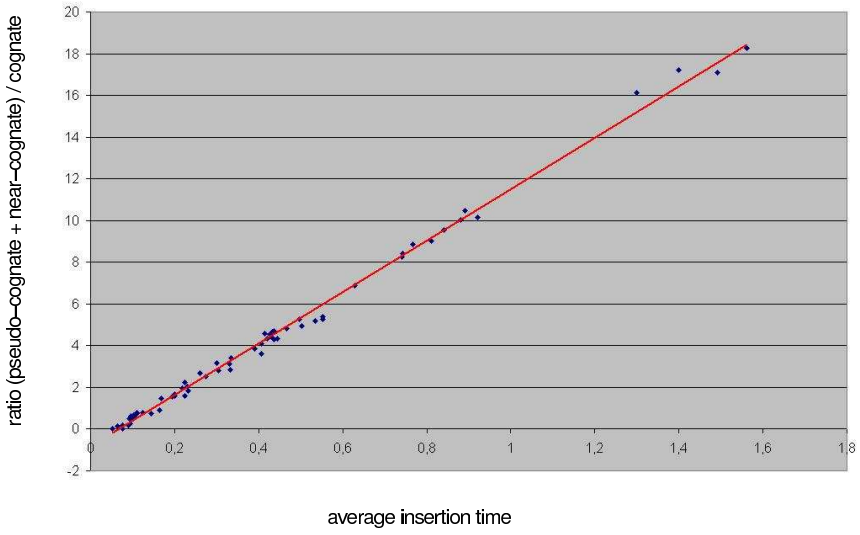

Figure 5: Correlation of $\frac{f_p+f_n}{f_c}$ ratio and average insertion times