Stochastic Kinetic Modeling of the Vesicular Stomatitis Virus (VSV)

by

Sebastian C. Hensel

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science
(Chemical Engineering)

at the

UNIVERSITY OF WISCONSIN-MADISON
2007
Stochastic Kinetic Modeling of the Vesicular Stomatitis Virus (VSV)

Sebastian C. Hensel, M.S.
Department of Chemical Engineering
University of Wisconsin-Madison, 2007
Professors John Yin and James B. Rawlings, Advisors

Viruses cause diseases such as influenza, AIDS and SARS, and thereby create a major and ever expanding global threat to human health. A better fundamental understanding of how viruses reproduce may enable the development of more effective anti-viral therapies. To reproduce, a virus must infect and utilize biosynthetic resources from a living host cell. As a model we study vesicular stomatitis virus (VSV), a relatively small virus whose processes of gene expression and genome replication are among the best characterized of all viruses. When cells are infected with a recombinant VSV that expresses green fluorescent protein (GFP), different cells infected by single virus particles produce different levels of GFP and different levels of virus, a finding that cannot be explained by an existing deterministic ODE kinetic model of VSV growth. We hypothesize that such distributions in growth behavior arise in part from fluctuations in the small levels of virus components present when a cell is initially infected. These components are then rapidly amplified by autocatalytic feedbacks in the reaction network that defines virus growth. A modeling approach that accounts for the stochastic firing of each reaction event would be appropriate, but stochastic approaches become computationally intractable as the number of species rapidly grows. We address this problem by developing a hybrid implementation of the model that combines a delayed stochastic simulation algorithm (DSSA) with Langevin equations to simulate the reactions that produce species in high numbers. Another problem is caused by fast switching species that have high production and even higher consumption rates. These species are highly reactive and fluctuate at low molecule levels and a quasi-steady state approximation (QSSA) has to be used to overcome the problem of small time steps caused by these species. Using our new modeling approach we find a reasonable match for the GFP molecule level from our model to the measured experimental results, which gives insight into the dynamics of GFP expression. The simulation results also show expected levels of genome replication and mRNA level and protein level rankings. Another more intriguing result is the phenomena of separation of population that corresponds to the lowest number of viral species, the L protein. These results are interesting because they are not just underlining experimental results, but also show features that have not yet been experienced. The separation of population is caused by the availability of the least abundant viral protein, which makes a stochastic description of the virus infection cycle unavoidable. Finally, because infections by different viruses generally involve the rapid amplification of small numbers of few components, we expect our approach to find broad application to the stochastic simulation of diverse viruses.
Acknowledgments

First of all, I would like to thank my research advisors Dr. John Yin and Dr. James Rawlings for their patience and support. Pursuing the same goal with two advisors has been a bigger challenge than expected. I am grateful to each of them for giving me the opportunity to study such a fascinating biological system and to learn about all the different simulation methods.

I would like to thank Dr. Yin for all the time we spent in discussion helping me to better understand the complexity of the biological system. His guidance and broad perspective helped me to gain new knowledge and to be open for new ideas.

I would like to thank Dr. Rawlings for teaching me about all the different simulation methods and for pushing me into the right direction. After almost every meeting I learned something new, and his constructive critique and feedback helped me to stay focused in my research.

I would like to thank Dr. Daniel Klingenberg and Dr. Sean Palecek for being part of my committee.

I also appreciate the helpful comments and the discussions with Dr. Thomas Kurtz, Dr. Daniel Anderson and Dr. Gheorge Craciun. The feedback and the new ideas in our meetings helped me to gain more insight and have a deeper understanding.

I would like to thank the past and current members of the Rawlings group and the members of the Yin group for their friendship and support. Special thanks go to Murali Rajamani for all his help in the beginning, and to Ethan Mastny for all the time and his patience in answering my questions. I am grateful to Kris Stauffer and Ying Zhu, who helped me to better understand the virus and to provide me with experimental data. I would also like to thank Paul Larsen, Marcus Reble and Rishi Amrit for their support and friendship.

Being from Germany I did not know what to expect of my time in Wisconsin. Getting used to all the differences in studying and living was an experience in itself. The two years I have spent in Madison have passed very quickly and meeting so many great people made my stay enjoyable and at the same time so memorable. I would like to thank Professor Matthias Zeitz and Professor Matthias Reuss for the opportunity and the support to study in Madison and to pursue a masters’s degree. I would also like to thank Stephan Deublein and Falk Eichhorn for their friendship and support. Sitting in the same boat helped a lot.

Special thanks also to Kimberly Koopman for her help and support, especially in the late stages of the thesis.

Finally, I give my thanks to my family for their steady support throughout my student life. Without their encouragement and guidance I would not be where I am today.
# Table of Contents

Abstract i

Acknowledgments ii

Table of Contents iii

List of Tables v

List of Figures vi

1 Introduction 1
   1.1 Stochastic Background .................................................. 1
   1.2 VSV Biology ................................................................. 2
   1.3 Experimental Methods and Results ...................................... 3

2 Reactions 6
   2.1 Transcription ............................................................... 6
   2.2 Translation ................................................................. 8
   2.3 Replication ................................................................. 10
   2.4 Host Factors ............................................................... 15

3 Implementation 16
   3.1 Delayed Stochastic Simulation Algorithm (DSSA) .................. 16
   3.2 The Langevin Equation .................................................... 17
   3.3 QSSA on the $N$ protein ................................................ 18
   3.4 Delay Replication Reaction .............................................. 20

4 Simulation strategy 23
   4.1 Hybrid Langevin implementation ....................................... 23

5 Results and Discussion 28

6 Conclusions 36
   6.1 Contributions ............................................................. 36
   6.2 Future Work ............................................................... 37

Bibliography 39

Appendix 42
List of Tables

2.1 Model parameters and delays of the VSV transcription reactions. . . . . . . . . 9
2.2 Model parameters of the VSV translation reactions. . . . . . . . . . . . . . . 10
2.3 Model parameters and delays of the VSV replication reactions. . . . . . . . 14
2.4 Host parameters of the BHK-21 cells. . . . . . . . . . . . . . . . . . . . . . . 15
3.1 List of mathematical letters, symbols, and abbreviations. . . . . . . . . . . . 22
4.1 List of mathematical letters, symbols, and abbreviations. . . . . . . . . . . . 27
A.1 Influence of the ribosomal footprint on the mean of the N protein level. . . 42
B.1 Model parameters for the QSSA example reaction network. . . . . . . . . . 44
List of Figures

1.1 Schematic reaction network of the Vesicular Stomatitis Virus (VSV). . . . . 4
2.1 Stochastic simulation run including all encapsidation (chain) reactions. . . . 13
5.1 Mean of genome levels over time. . . . . . . . . . . . . . . . . . . . . . . . 29
5.2 Mean of mRNA levels over time. . . . . . . . . . . . . . . . . . . . . . . . . 30
5.3 Mean of protein levels over time. . . . . . . . . . . . . . . . . . . . . . . . 30
5.4 Comparison of the mean GFP molecule level of the model to the measurements. 31
5.5 Distribution of the (-)RNA genome level . . . . . . . . . . . . . . . . . . . 32
5.6 Dependence between (-)RNA genome level and L mRNA level. . . . . . . . 33
5.7 Distribution of the GFP molecule level . . . . . . . . . . . . . . . . . . . . 34
B.1 Three representative simulation runs of the full model. . . . . . . . . . . . . 47
B.2 Three representative simulation runs of the QSSA model. . . . . . . . . . . . 47
B.3 Comparison of the full model probability distribution to the calculated QSSA probability distribution. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
Chapter 1

Introduction

1.1 Stochastic Background

In the early states of a viral infection, when molecule levels are low, random events dramatically influence the outcome of a viral infection. Past deterministic kinetic models of VSV growth account for the production, interactions and decay of essential VSV molecular species, including: VSV mRNA, genomic and anti-genomic full-length RNA, VSV proteins, several protein-RNA intermediates, and VSV progeny precursors; the model captures essential relationships among virus constituents [1]. Experiments have revealed broad distributions of virus-mediated gene expression measured at the single-cell level by flow cytometry, as well as distributions of virus yields measured from single infected cells, features that deterministic models cannot explain. To better understand mechanistically how such distributions may come about, a stochastic VSV infection model has to be developed that allows for inherent fluctuations in the levels of viral constituents. Such effects will be most significant during the earliest stages of infection when levels of viral species are low; these effects may cause extinction of virus species from infected host cells or contribute to distributions in virus progeny production. For low numbers of molecules, the continuous description of the system is not valid, because the molecule numbers are integer values and reactions cause integer jumps in these values. These systems are typically modeled as discrete jump Markov processes. The stochastic simulation algorithm (SSA), also know as the Gillespie Algorithm [2], is an exact simulation method for these Markov processes. Another simulation algorithm for the Markov jump processes is the next reaction method, which has been refined by Gibson and Bruck to reduce the required number of random numbers [3]. The delayed stochastic
simulation algorithm (DSSA) is an extended version of the SSA. It accounts for delays that are involved in various reactions, including gene transcription and replication reactions, as well as protein translation reactions [4] [5]. Other methods exist that directly analyze the master equation and approximation methods based on the Fokker-Planck or Langevin equations [6] [7]. Several hybrid models, that are based on the separation of time scales between fast and slow reactions, are used to reduce the computational effort of the full stochastic models [8] [9]. Stochastic models have been developed for HIV and phage λ that explain the diversity of phenotypes or pathway bifurcations, features that could not be explained with a deterministic model, while the difference in stochastic and deterministic modeling of viral kinetics has also been studied [10] [11] [12]. The challenge of this work is to implement a stochastic model of a full virus infection that initially has a low number of molecules. While some species are growing rapidly, others stay at low numbers and are produced and consumed with high reaction rates and are also involved in delayed reactions. These features cannot be handled by a simple simulation algorithm; the model has to be implemented with a simulation strategy that uses different implementation methods for different conditions. This work will focus on how to develop a detailed stochastic simulation strategy for the life cycle of the Vesicular Stomatitis Virus (VSV).

1.2 VSV Biology

The Vesicular Stomatitis Virus (VSV) is a single negative strand RNA virus ((-)ssRNA). It is a member of the Rhabdoviridae family, which includes the rabies virus. Its molecular processes are among the best characterized of all viruses; it grows rapidly and is non-pathogenic to humans. VSV is currently being developed as a vaccine in which it is used as a parasite to other viruses and as an oncolytic therapy. Each virus particle carries a single 11-kb (-)RNA genome that encodes five genes. Every gene encodes exactly one protein, the nucleocapsid
protein N, the phospho protein P, the matrix protein M, glyco protein G and the large polymerase protein L. The lattermost protein is used to transcribe the genome into its messages (mRNAs). These viral messages are then translated into their proteins using host ribosomes. Every protein has specific functions that are necessary to create viable virus progeny particles. The N protein encapsidates the genome and stabilizes it, while at the same time it is responsible for the switch between transcription and replication. Once the (-)RNA genome is fully encapsidated, it serves as a template to create the anti-genome ((+)RNA)). The (+)RNA strand also has to be encapsidated by the N proteins in order to serve as a template to create the (-)RNA genome. The P protein has its part in the encapsidation process of the genome, as well as in the transcription and replication reactions. The G protein forms spikes at the membrane of the host cell, which will then be used for the viral membrane. The G spikes on the outside of the virus particles will help to attach to other cells that can be infected. The M protein is part of the inner membrane of the virus, and is also responsible for shutting down the host and the viral translation. Although the P, M and G proteins have distinct roles in the virus infection cycle, this work does not include the effects of these proteins in the virus model. One VSV particle consists of one (-)RNA genome strand, 1,258 N, 466 P, 1,826 M, 1,205 G and 50 L proteins. This will also be the initial condition for our simulations.

### 1.3 Experimental Methods and Results

The following reaction network is based on the recombinant strand that has a GFP insert after the G-Protein gene (G-GFP), 3’–N–P–M–G–GFP–L–5’ as it can be seen in Fig. 1.1. The GFP protein stands for “Green Fluorescent Protein” and measurements of its intensity are linked to its molecular protein level. The experimental results are based on an infection with a multiplicity of infection (MOI) of one, which means that each cell was infected with
Figure 1.1: This plot shows the genome of the VSV virus that we are using in our simulations, a schematic of the VSV reaction network and descriptions of the more complex reactions. (-)RNA and (+)RNA genomes include the naked and the partially encapsidated strands, but not the fully encapsidated strands. (I) shows the encapsidation reaction of the (-)RNA genome. The (+)RNA encapsidation follows the same procedure. In (II) we can see how the replication of the encapsidated (-)RNA genome consumes one L protein at initiation and how the (+)RNA template is formed. The L protein will then be released with the complete (+)RNA strand. The synthesis of the (-)RNA strand follows the same mechanistics. (III) shows how the mRNAs are formed from the naked or partially encapsidated (-)RNA genome. The L protein can be released at each gene junction, which leads to different ratios of mRNAs.
an average of one virus particle. In our experiments we use two different recombinant virus strains, the G-GFP and the GFP-N strain. G-GFP has the GFP gene in the fourth position of the genome, while GFP-N has the GFP gene at the first position, closer to the 3’ end of the genome. The host cells we use for our infection experiments are baby hamster kidney cells (BHK-21). All experiments and data have been performed and provided by Ying Zhu, a fellow member of the John Yin group.

Baby hamster kidney (BHK-21) cells were originally obtained from Dr. I. Novella (Medical College of Ohio). No antibiotics were used and cells were subcultured no more than 100 passages to minimize the artifacts due to cell senescence. Viability of cell populations always approached 100% at the time of experiments. In all the experiments performed in this paper, BHK cells were plated in 6-well culture plates one day before infection.

Recombinant vesicular stomatitis virus that expresses green fluorescence protein at the fourth position after 3’ leader (G-GFP strain) was generously provided by Dr. G. Wertz (University of Virginia). Generally, it was recovered from cDNA clones that had been engineered to express an additional transcriptional unit, purified and amplified in BHK cells as described by Whelan et al. [13]. Virus infectivity titers were expressed as plaque forming unit (PFU) per ml. BHK cells were infected with the G-GFP strain at multiplicities of infection (MOIs) of 5, 1, 0.1 and 0.01. The inoculum was removed after one hour of adsorption. Therefore, the time of initiation of infection was not tightly synchronized at indicated times post infection, cells were released and settled on ice. The cells were filtered right before the analysis.

Detection and quantification of GFP signal was performed using a MoFlo cell sorter (Dako-Cytomation, Fort Collins, CO) with Summit software version 4.13 under biosafety level-3 (BL-3) conditions. The gate for the definition of GFP-positive cells was positioned such that 0.1% of the cells without expressing GFP were positive. Data was acquired based on gated cells, using Flowjo 7.1 software (Tree Star, Inc).
Chapter 2

Reactions

The full virus life cycle consists of various types of reactions. This model includes transcription of the genome into its mRNAs (messages, templates for virus proteins), translation of the messages into their proteins, and replication reactions to synthesize the positive and negative RNA strand genomes. The transcription reactions of the genome into its mRNAs are delayed reactions, while the translation reactions of the mRNAs into proteins are non-delayed. It also includes a chain reaction to encapsidate the (+)RNA and (-)RNA genome, and the replication reaction of the fully encapsidated genome.

Every transcription reaction involves a different delay. They are initiated at one point in time but the products appear much later, with delays ranging from a couple of minutes up to over an hour. These reactions are modeled with a delayed stochastic simulation algorithm (DSSA). Every transcription reaction consumes one polymerase (L protein) at initiation and releases it upon completion. The translation reactions are much faster than the transcription reactions and are therefore modeled with the simpler stochastic simulation algorithm (SSA). The replication reactions consist of non-delayed chain reactions to encapsidate the (+) and (-)RNA genome and a delayed replication reaction that consumes the polymerase at initiation and releases it upon completion of the process. All the following reactions that involve a delay will be marked with a (*). All reactions are modeled as irreversible reactions with reaction rates calculated through mass action kinetics.

2.1 Transcription

The transcription reactions transcribe all genes of the genome into their mRNAs, which serve as templates for the viral proteins.
The (-)RNA in this equation stands for all (-)RNA genomes that are not fully encapsidated. The model parameters and delays are given in Table 2.1. All six transcription reactions involve different delays. The delays have been calculated by dividing the nucleotide length of the genes by the $L$ polymerase elongation rate [14]. While $L$ is released with the last mRNA, messages more closely positioned to the 3’ transcription initiation site of the genome are released earlier in time. The initiation rates are calculated by multiplying the different attenuation factors $\Phi$, [15] [14] [16], with the transcription initiation rate $k_1^*$, [17]. The $L$ protein does not always read through the whole genome, but stops at intergenic regions. The attenuation factors reflect the probability of the $L$ protein transcribing until it reaches the intergenic regions after each gene, where it falls off. The nucleic acids $S_1$ are assumed to be in abundance. The transcription reactions have the following rate expressions.
\[ r_{1(1)} = k_{1(1)}(-)RNA \cdot L \]  \hspace{1cm} (2.2a)
\[ r_{1(2)} = k_{1(2)}(-)RNA \cdot L \]  \hspace{1cm} (2.2b)
\[ r_{1(3)} = k_{1(3)}(-)RNA \cdot L \]  \hspace{1cm} (2.2c)
\[ r_{1(4)} = k_{1(4)}(-)RNA \cdot L \]  \hspace{1cm} (2.2d)
\[ r_{1(5)} = k_{1(5)}(-)RNA \cdot L \]  \hspace{1cm} (2.2e)
\[ r_{1(6)} = k_{1(6)}(-)RNA \cdot L \]  \hspace{1cm} (2.2f)

### 2.2 Translation

The translation reactions produce proteins by translating the messages (mRNAs) using the host cell ribosomes:

\[ N_{mRNA} + S_2 \xrightarrow{k_2} N_{mRNA} + N_{Protein} \]  \hspace{1cm} (2.3a)
\[ P_{mRNA} + S_2 \xrightarrow{k_2} P_{mRNA} + P_{Protein} \]  \hspace{1cm} (2.3b)
\[ M_{mRNA} + S_2 \xrightarrow{k_2} M_{mRNA} + M_{Protein} \]  \hspace{1cm} (2.3c)
\[ G_{mRNA} + S_2 \xrightarrow{k_2} G_{mRNA} + G_{Protein} \]  \hspace{1cm} (2.3d)
\[ GFP_{mRNA} + S_2 \xrightarrow{k_2} GFP_{mRNA} + GFP_{Protein} \]  \hspace{1cm} (2.3e)
\[ L_{mRNA} + S_2 \xrightarrow{k_2} L_{mRNA} + L_{Protein} \]  \hspace{1cm} (2.3f)

The parameters for the translation reactions are given in Table 2.2. The translation rate constant \( k_2 \) is the same for all translation reactions as the ribosomal elongation rate is the same for all mRNAs. It is calculated by dividing the translational elongation rate by the
Table 2.1: Model parameters and delays of the VSV transcription reactions.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Φ [15] [14] [16]</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiation rate constant [17]</td>
<td>$k_1^*$</td>
<td></td>
<td>0.0461 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1a) rate constant</td>
<td>$k_{1(1)}$</td>
<td>1.0</td>
<td>0.0461 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1b) rate constant</td>
<td>$k_{1(2)}$</td>
<td>0.75</td>
<td>0.0346 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1c) rate constant</td>
<td>$k_{1(3)}$</td>
<td>0.5625</td>
<td>0.0259 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1d) rate constant</td>
<td>$k_{1(4)}$</td>
<td>0.422</td>
<td>0.0195 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1e) rate constant</td>
<td>$k_{1(5)}$</td>
<td>0.422</td>
<td>0.0195 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.1f) rate constant</td>
<td>$k_{1(6)}$</td>
<td>0.0633</td>
<td>0.0029 $\frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Polymerase elongation rate [14]</td>
<td>$k_{e,p}$</td>
<td></td>
<td>3.7 $\frac{nt}{s}$</td>
</tr>
<tr>
<td>Length of $N$ gene [16]</td>
<td>$l_N$</td>
<td></td>
<td>1,333 $nt$</td>
</tr>
<tr>
<td>Length of $P$ gene [16]</td>
<td>$l_P$</td>
<td></td>
<td>822 $nt$</td>
</tr>
<tr>
<td>Length of $M$ gene [16]</td>
<td>$l_M$</td>
<td></td>
<td>838 $nt$</td>
</tr>
<tr>
<td>Length of $G$ gene [16]</td>
<td>$l_G$</td>
<td></td>
<td>1,672 $nt$</td>
</tr>
<tr>
<td>Length of $GFP$ gene [16]</td>
<td>$l_{GFP}$</td>
<td></td>
<td>720 $nt$</td>
</tr>
<tr>
<td>Length of $L$ gene [16]</td>
<td>$l_L$</td>
<td></td>
<td>6,380 $nt$</td>
</tr>
<tr>
<td>Reaction (2.1a) delay</td>
<td>$\tau_{t1}$</td>
<td></td>
<td>600.27 s</td>
</tr>
<tr>
<td>Reaction (2.1b) delay</td>
<td>$\tau_{t2}$</td>
<td></td>
<td>1062.4 s</td>
</tr>
<tr>
<td>Reaction (2.1c) delay</td>
<td>$\tau_{t3}$</td>
<td></td>
<td>1528.9 s</td>
</tr>
<tr>
<td>Reaction (2.1d) delay</td>
<td>$\tau_{t4}$</td>
<td></td>
<td>2220.8 s</td>
</tr>
<tr>
<td>Reaction (2.1e) delay</td>
<td>$\tau_{t5}$</td>
<td></td>
<td>2655.4 s</td>
</tr>
<tr>
<td>Reaction (2.1f) delay</td>
<td>$\tau_{t6}$</td>
<td></td>
<td>4619.7 s</td>
</tr>
</tbody>
</table>
Table 2.2: Model parameters of the VSV translation reactions.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ribosome elongation rate [18]</td>
<td>$k_{e,r}$</td>
<td>$18 \frac{nt}{sec}$</td>
</tr>
<tr>
<td>Ribosome footprint [1]</td>
<td>$s_{rib}$</td>
<td>$238.5 nt$</td>
</tr>
<tr>
<td>Translation rate constant</td>
<td>$k_2$</td>
<td>$0.07551 \frac{1}{s}$</td>
</tr>
</tbody>
</table>

Ribosome footprint [18] [1]. The ribosome footprint is a parameter that has been estimated in our previous deterministic model [1] by fitting it to four independent sets of data from the literature and own experimental data. A discussion on the sensitivity of the parameter can be found in Appendix A. The ribosomes ($S2$) are assumed to be unlimited host resources. The delays are much smaller than the delays for the transcription, and therefore are negligible. The translation reactions have the following rate expressions.

$$
\begin{align*}
  r_{2(1)} &= k_2 N_{mRNA} \\
  r_{2(2)} &= k_2 P_{mRNA} \\
  r_{2(3)} &= k_2 M_{mRNA} \\
  r_{2(4)} &= k_2 G_{mRNA} \\
  r_{2(5)} &= k_2 GFP_{mRNA} \\
  r_{2(6)} &= k_2 L_{mRNA}
\end{align*}
$$

2.3 Replication

The replication consists of four different types of reactions. In order to synthesize a copy of the (-)-strand RNA genome, we need all four reactions to happen in the following order.
The (-)strand RNA has to be encapsidated with 1258 $N$ proteins before it can serve as a template for (+)strand RNAs via polymerase replication. The (+)strand RNA also has to be encapsidated by 1258 $N$ proteins in order to serve as a template to synthesize a naked (-)strand RNA. If the encapsidation reactions are all modeled as single chain reactions, the large number of reactions would pose a big computational burden for the full stochastic model.

\[
\begin{align*}
(-)RNA + N & \xrightarrow{k_3} (-)RNA_1 \\
(-)RNA_1 + N & \xrightarrow{k_3} (-)RNA_2 \\
& \quad \cdots \xrightarrow{k_3} \cdots \\
(-)RNA_{1257} + N & \xrightarrow{k_3} (-)RNA_{1258} \\
(-)RNA_{1258} + L & \xrightarrow{k_3} (-)RNA_{1258} + (+)RNA + L \quad (*) \\
(+){RNA} + N & \xrightarrow{k_3} (+)RNA_1 \\
(+)RNA_1 + N & \xrightarrow{k_3} (+)RNA_2 \\
& \quad \cdots \xrightarrow{k_3} \cdots \\
(+)RNA_{1257} + N & \xrightarrow{k_3} (+)RNA_{1258} \\
(+)RNA_{1258} + L & \xrightarrow{50-k_3} (+)RNA_{1258} + (-)RNA + L \quad (*) \quad (2.5)
\end{align*}
\]

The exact stochastic model would include the simulation of all chain reactions and all species that take part in it. The simulation of this model would cost a lot of memory to store all the states and it would also be computationally very expensive. Fig. 2.1 shows one simulation run that includes all chain reactions. (-)RNA genomes includes all naked and partially encapsidated (-)RNA strands, but not the fully encapsidated (-)RNA strands. The simulation has been stopped at 100,000 iterations. The full infection cycle cannot be simulated with the
model used for this simulation run, because of the fast switching species in the system. It can be seen that all the production and consumption reactions of the $N$ protein dictate the step size, while its level fluctuates intensively. In order to reduce the computational burden of the encapsidation reaction, future work has to be done in order to approximate chain reactions in which the level of substrates fluctuates over time. However this is not the focus in this work. Our model is based on the full model until the $N$ protein begins to switch fast. After that, all chain reactions are implemented using only one single delayed reaction for the full encapsidation of the genome, while the delay is calculated using the mean of the QSSA distribution of the $N$ protein that we will describe later. The encapsidation reaction of this model has the following form.

\begin{align}
(\text{-})RNA + N & \xrightarrow{k_2} (\text{-})RNA_{1258} \quad (*) \tag{2.6a} \\
(\text{-})RNA_{1258} + L & \xrightarrow{k_4} (\text{-})RNA_{1258} + (\text{+})RNA + L \quad (*) \tag{2.6b} \\
(\text{+})RNA + N & \xrightarrow{k_2} (\text{+})RNA_{1258} \quad (*) \tag{2.6c} \\
(\text{+})RNA_{1258} + L & \xrightarrow{k_5} (\text{+})RNA_{1258} + (\text{-})RNA + L \quad (*) \tag{2.6d}
\end{align}

The replication parameters and delays are given in Table 2.3. The initiation of the $(\text{+})$RNA synthesis reaction and all chain reactions have the same rate constant as the transcription initiation rate constant $k_1^\text{\text{+}}$. It has been found that the promoter strength of the fully encapsidated $(\text{+})$RNA strand is higher than the strength of the negative strand. The $(\text{+})$RNA strand can be found in a level that is up to 50 times higher than the level of the $(\text{-})$RNA strand. Therefore, I set the initiation reaction rate constant for the $(\text{-})$RNA synthesis reaction to a value 50 times higher than the rate constant of the $(\text{+})$RNA synthesis. The total time needed for the genome synthesis is almost an hour and it is calculated by dividing the total genome nucleotide length by the $L$ polymerase elongation rate [14]. The delay time
Figure 2.1: This plot shows the (+)RNA and (-)RNA genome levels and the N mRNA and N protein levels over the course of one simulation run that includes all chain reactions described in Eq. 2.5. The first drop (I) in the N protein level is the first full encapsidation reaction described in (I) in Fig. 1.1. The delay before the first N mRNA occurs is the time required to produce transcripts from the genome, which has been explained in (III) in Fig. 1.1. The time between (I) and (III) is the time it takes to synthesize the whole genome, which has been described in (II) in Fig. 1.1. It can be seen that the N protein is highly reactive and begins to fluctuate intensively right after (II), when the first (+)RNA genomes are replicated, which consume the N protein. (-)RNA genomes and (+)RNA genomes include the naked and the partially encapsidated strands, but not the fully encapsidated genomes.
Table 2.3: Model parameters and delays of the VSV replication reactions.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of genome [16]</td>
<td>$l_g$</td>
<td>$11,765 \text{ nt}$</td>
</tr>
<tr>
<td>Polymerase elongation rate [14]</td>
<td>$k_{e,p}$</td>
<td>$3.7 \text{ nt/s}$</td>
</tr>
<tr>
<td>Reaction (2.6a,2.6b) rate constant [17]</td>
<td>$k_3$</td>
<td>$0.0461 \frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.6b) rate constant [17]</td>
<td>$k_4$</td>
<td>$0.0461 \frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.6d) rate constant [17]</td>
<td>$k_5$</td>
<td>$2.305 \frac{1}{\text{sec}}$</td>
</tr>
<tr>
<td>Reaction (2.6a) delay</td>
<td>$\tau_{r_1}$</td>
<td>varies</td>
</tr>
<tr>
<td>Reaction (2.6b) delay</td>
<td>$\tau_{r_2}$</td>
<td>$3179.7 \text{ s}$</td>
</tr>
<tr>
<td>Reaction (2.6c) delay</td>
<td>$\tau_{r_3}$</td>
<td>varies</td>
</tr>
<tr>
<td>Reaction (2.6d) delay</td>
<td>$\tau_{r_4}$</td>
<td>$3179.7 \text{ s}$</td>
</tr>
</tbody>
</table>

for the encapsidation reaction varies with the total number of genomes and the mean of the QSSA distribution of the $N$ protein. There are methods to calculate the delay time via a gamma distribution of multiple reaction events if the reaction rate is not changing over time. However, this is not the case for the chain reaction rate in this model, where the changes in the highly reactive protein $N$ and the genome level cause fluctuating reaction rates. The method to calculate the delay and the QSSA distribution of the $N$ protein level will be discussed later. The replication reactions have the following rate expressions.

\[
r_{3(1)} = k_3(-)RNA \cdot N \quad (2.7a)
\]
\[
r_{3(2)} = k_4(-)RNA_{1258} \cdot L \quad (2.7b)
\]
\[
r_{3(3)} = k_3(+)RNA \cdot N \quad (2.7c)
\]
\[
r_{3(4)} = k_5(+)RNA_{1258} \cdot L \quad (2.7d)
\]
2.4 Host Factors

Among all host factors that could influence our reaction network, we only focus on one host factor in this model. The cell size, as well as the amount of host resources that can be used by the virus varies from cell cycle to cell cycle. In this model, the cell size is assumed to be constant during the infection cycle and it is given in Table 2.4. The shape of the virus is assumed to be spherical and we have measured the average diameter of the BHK cells to calculate the volume of the host cells. Reaction rates depending on concentrations that have been derived from experimental data is converted into molar reaction rates using this average host cell volume.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average cell diameter</td>
<td>$d$</td>
<td>$16\mu m$</td>
</tr>
<tr>
<td>Average cell volume</td>
<td>$v$</td>
<td>$2.14e^{-15}l$</td>
</tr>
</tbody>
</table>
Chapter 3

Implementation

3.1 Delayed Stochastic Simulation Algorithm (DSSA)

The model is implemented with a delay stochastic simulation algorithm (DSSA) that has already been developed by Bratsun [5] and Barrio [4]. The delays have to be treated carefully because all delayed reactions consume species upon initiation and release several products at different points in time. The DSSA handles delayed and non-delayed reactions, and is an extended version of the original Gillespie Algorithm [2]. The following algorithm uses waiting and delay times, and also delayed reactions that change the state of the system at both initiation and completion. \( \nu_i \) is the stoichiometric matrix of the non-delayed reactions and the initiation of delayed reactions, while \( \nu_d \) is the stoichiometric matrix of the completion of the delayed reactions. All stored reaction times are saved in \( T_d \).

1. Set time \( t \) equal to zero, the number of species \( x \) to \( x_0 \) and the first stored completion time \( t_d \) to \( \infty \)

2. Calculate all \( m \) reaction rates, \( r_j(x) = k_j a_j(x) \)

3. Calculate the total reaction rate, \( r_{tot} = \sum_{j=1}^{m} r_j \)

4. Generate two random numbers \((p_1, p_2)\) uniformly distributed on \((0,1)\)

5. Calculate the stochastic time step \( \Delta t = - \ln (p_1)/r_{tot} \)

6. if there is a stored reaction \( n \) to finish in \([t, t + \Delta t)\):
   - Discard steps 4-5 and update time \( t = \text{min}(T_d) = t_d \)
   - Update species number \( x = x + \nu_d(n) \)
• Repeat steps 2-7 while \( t \leq t_{\text{final}} \)

7. else:

• Find reaction \( n \), such that \( \sum_{j=1}^{n-1} r_j(x) < p_2 r_{\text{tot}} \leq \sum_{j=n}^{m} r_j(x) \)

• Update time \( t = t + \Delta t \)

• Update species number \( x = x + \nu_{i(n)} \)

• If reaction is delayed, store that at time \( t+\tau \) the system must be updated according to the completion of reaction \( n \)

• Repeat steps 2-7 while \( t \leq t_{\text{final}} \)

### 3.2 The Langevin Equation

First runs revealed (not shown) that there are species produced in millions of molecules. In order to simulate faster, an approximation has to be used that allows for taking bigger time steps, but still accounts for fluctuations. A method that has already been explored in various stochastic models is the use of Langevin equations. The formulation of the chemical Langevin equation has been addressed by Gillespie [7]. The Langevin equation is a good approximation under certain conditions that may change during a simulation run.

\[
x(i + 1) = x(i) + r \Delta t + \sqrt{r \Delta t} R, \quad R \sim N(0, 1)
\]  
\[ (3.1) \]

The equation above characterizes a first order approximation of a continuous time stochastic process, in which \( x(i+1) \) is the next state, \( x(i) \) is the initial state, \( r \Delta t \) is the first order change in \( x(i) \), and \( \sqrt{r \Delta t} R \) is the standard deviation of that first order change, multiplied with a normally distributed random number \( R \). Therefore, the next state is not only calculated by the mean of the reaction rate, but by a normally distributed probability distribution.
around that mean. This equation is only valid when the number of all molecules that are influenced by a reaction is high, and the exponential distribution of the reaction rate can be approximated by a normal distribution. Then the reaction can be modeled with a Langevin equation and is not updated as part of the DSSA. The model has to be capable of switching between the implementation via the Langevin equations and the DSSA.

3.3 QSSA on the $N$ protein

The $N$ protein is a highly reactive state in the system. It is switching between low values of $N$ such that they cannot be approximated by the Langevin equations. The $N$ protein is produced by its mRNA with a high rate, while it is consumed by both positive and negative RNA genomes. A simpler model that captures the encapsidation reaction and the essential rapid switching behavior of $N$ (between high and low molecule values) is discussed in Appendix B. All rates for the chain reactions are calculated by the same reaction rate constant $k_3$. The $N_{mRNA}$ and the genomes are present in large amounts, while $N$ is highly reactive. Considering the $N_{mRNA}$ and the genome levels to be constant at some values over some interval of interest, the master equation for $N$ can be written as follows:

$$
\frac{dP(N,t)}{dt} = -k_2 N_{mRNA} P_N + k_2 N_{mRNA} P_{N-1} \\
-k_3 \sum_{i=0}^{1257} ((-)^{RNA_i} + (+)^{RNA_i}) NP_N \\
+k_3 \sum_{i=0}^{1257} ((-)^{RNA_i} + (+)^{RNA_i})(N + 1)P_{N+1}
$$

(3.2)

$P_N$ is shorthand for $P(N,t)$. If the production and consumption rates are high such that $N$ equilibrates to its steady-state condition on a fast time scale compared to the evolution of
level mRNA and the genomes, the steady-state probability density of $N$ can be found by setting $dP(N,t)/dt = 0$. Using

$$r_1 = k_2 N_{mRNA}$$

$$r_2 = k_3 \sum_{i=0}^{1257} ((-)^{RNA_i} + (+)^{RNA_i})$$

(3.3)

the following equation can be derived:

$$0 = -r_1 P_N + r_1 P_{N-1} - r_2 N P_N + r_2 (N+1) P_{N+1}$$

(3.4)

Evaluating this equation for $N = 0, 1, 2, ...$

$$N = 0 \quad 0 = -r_1 P_0 + r_2 P_1$$

$$N = 1 \quad 0 = -r_1 P_1 + \underbrace{r_1 P_0 - r_2 P_1}_{\text{zero from } N = 0} + 2r_2 P_2$$

$$N = 2 \quad 0 = -r_1 P_2 + \underbrace{r_1 P_1 - 2r_2 P_2}_{\text{zero from } N = 1} + 3r_2 P_3$$

$$\ldots \quad \ldots$$

$$N = n \quad 0 = -r_1 P_n + \underbrace{r_1 P_{n-1} - nr_2 P_n}_{\text{zero from } N = n-1} + (n+1)r_2 P_{n+1}$$

(3.5)

This relation provides the following recursion in terms of $P_0$.

$$P_N = \frac{1}{N!} \alpha^N P_{N-1}$$

$$P_N = \frac{1}{N!} \alpha^N P_0 \quad \alpha = \frac{a}{b} = \frac{k_2 N_{mRNA}}{k_3 \sum_{i=0}^{1257} ((-)^{RNA_i} + (+)^{RNA_i})}$$

(3.6)
Summing $P_N$ over $N$ gives us:

$$\sum_{N=0}^{\infty} P_N = (1 + \frac{\alpha}{1!} + \frac{\alpha^2}{2!} + \frac{\alpha^3}{3!} + \cdots) P_0$$

$$1 = e^\alpha P_0$$

$$P_0 = e^{-\alpha}$$  \hspace{1cm} (3.7)

The quasi-steady probability density of $N$ is therefore:

$$P_N = \frac{1}{N!} \alpha^N e^{-\alpha}$$  \hspace{1cm} (3.8)

### 3.4 Delay Replication Reaction

As mentioned earlier, the encapsidation reaction of the genome is modeled by a delayed reaction. The reaction rate constant $k_3$ for all chain reactions is the same. When the distribution of the $N$ protein stays constant over the amount of time that it takes to encapsidate a whole genome, the average reaction rate for a single chain reaction is the one calculated by using only the mean of the QSSA distribution that has just been derived. This feature is also shown in the simpler example in B. All chain reactions follow the same reaction rate and time, and summing over all reaction times, the total time needed to encapsidate the whole genome can be calculated. Please note that the delay we use is only an approximation and does not reflect the whole stochasticity of all encapsidation reactions. However, it gives us the possibility to simulate further in time, without firing all single chain reactions with small time steps. In the full model all chain reactions can now be modeled as a delayed reaction that is initiated by the first chain reaction with the $N$ protein level drawn from the QSSA probability density of $N$. All delayed chain reactions are stored in $T_{d,rep}$. When the distribution of $N$ changes, and therefore the reaction rate and the delay time changes, the
remaining time of each stored delayed encapsidation reaction has to be updated accordingly.

This method will be described later.
Table 3.1: List of mathematical letters, symbols, and abbreviations. Listed in order of appearance.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSSA</td>
<td>Delayed Stochastic Simulation Algorithm</td>
</tr>
<tr>
<td>$\nu_i$</td>
<td>Stochiometric matrix of immediate reactions</td>
</tr>
<tr>
<td>$\nu_d$</td>
<td>Stochiometric matrix of delayed reactions</td>
</tr>
<tr>
<td>$T_d$</td>
<td>Vector of all stored delayed reaction completion times</td>
</tr>
<tr>
<td>$t_d$</td>
<td>Earliest stored completion time of delayed reactions</td>
</tr>
<tr>
<td>$t$</td>
<td>Simulation time</td>
</tr>
<tr>
<td>$x$</td>
<td>Vector of species numbers</td>
</tr>
<tr>
<td>$x_0$</td>
<td>Initial condition of species numbers</td>
</tr>
<tr>
<td>$r_j(x)$</td>
<td>Reaction rate of reaction $j$</td>
</tr>
<tr>
<td>$k_j$</td>
<td>Rate constant of reaction $j$</td>
</tr>
<tr>
<td>$k_ja_j(x)$</td>
<td>Reaction rate of reaction $j$</td>
</tr>
<tr>
<td>$r_{tot}$</td>
<td>Total reaction rate</td>
</tr>
<tr>
<td>$p$</td>
<td>Uniformly distributed random number on $(0,1)$</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>Time step</td>
</tr>
<tr>
<td>$n$</td>
<td>Next reaction or row of stochiometric matrix $\nu_i$ or $\nu_d$</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Reaction delay</td>
</tr>
<tr>
<td>$t_{final}$</td>
<td>Length of simulation</td>
</tr>
<tr>
<td>$x(i)$</td>
<td>State at iteration $i$</td>
</tr>
<tr>
<td>$r$</td>
<td>Reaction rate</td>
</tr>
<tr>
<td>$R$</td>
<td>Normally distributed random number with $\mu = 0$ and $\sigma = 1$</td>
</tr>
<tr>
<td>QSSA</td>
<td>Quasi-Steady-State Approximation</td>
</tr>
<tr>
<td>$P(N,t)$</td>
<td>Probability of $N$ being in state $N$ at time $t$</td>
</tr>
<tr>
<td>$P_N$</td>
<td>Shorthand for $P(N,t)$</td>
</tr>
<tr>
<td>$r_1$</td>
<td>Shorthand for $k_2N_mRNA$</td>
</tr>
<tr>
<td>$r_2$</td>
<td>Shorthand for $k_3\sum_{i=0}^{1257}((-)RNA_i + (+)RNA_i)$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Mean of the OSSA probability density of the $N$ protein</td>
</tr>
<tr>
<td>$T_{d,rep}$</td>
<td>Vector of all stored encapsidation reaction completion times</td>
</tr>
</tbody>
</table>
Chapter 4

Simulation strategy

The VSV model in this work consists of a variety of different delayed and non-delayed reactions and cannot be simulated by only using the DSSA because the computational burden of the fast reactions is too high. In order to lower the computation time, we have to apply several approximations that were discussed earlier. The simulation strategy focuses on how to combine all the different methods and approximations for a computationally inexpensive and still exact stochastic simulation of the model.

4.1 Hybrid Langevin implementation

This section focuses on the separation of reaction subsets. In recent methods, systems have been partitioned into subsets of different time scales, or fast and slow reaction subsets [8] [9]. These methods can only be applied when molecule levels are high, but the methods do not work accurately for fast switching states at low levels. In this work, reactions that influence molecules at high numbers are separated from reactions that influence molecules at low numbers. The “high” molecule number reactions are approximated via Langevin equations, while the “low” molecule number reactions are treated as stochastic events. For this model, the molecule level that separates “high” and “low” molecule number reactions is set to $n_{sw} = 100$, which divides all reactions into $m_1$ “low” molecule number reactions and $m_2$ “high” molecule number reactions. The maximum Langevin step is set to a value that we do not change more than one percent, $tol = 0.01$. When the stochastic step is bigger than the maximum Langevin step, only the continuous Langevin states are updated and the stochastic time step is discarded. This is valid because the reaction rates are exponentially distributed and therefore memoryless. Using this separation, the following hybrid Langevin
algorithm can be proposed:

1. Set time $t$ equal to zero, the number of species $x$ to $x_0$ and define $n_{sw}$

2. Separate the reactions into “high” and “low” molecule reactions by comparing $n_{sw}$ to all non-zero stoichiometric species of each reaction

3. Round all non-integer molecule levels that are smaller than $n_{sw}$

4. Calculate all “low” molecule number reaction rates, $r_l = k_l a_l(x)$ and the “high” molecule number reaction rates, $r_h = k_h a_h(x)$

5. Calculate the total “low” molecule number reaction rate, $r_{tot} = \sum_{l=1}^{m_1} r_l$

6. Generate two random numbers $(p_1, p_2)$ uniformly distributed on (0,1)

7. Calculate the stochastic time step $\Delta t = -\ln(p_1)/r_{tot}$

8. Calculate the maximum Langevin step $\Delta h t = \min(tol \cdot \min(abs(x_h/r_h)))$

9. if $\Delta t \leq \Delta h t$
   
   - Set time step $\Delta t = \Delta t$ and update $t = t + \Delta t$
   
   - Find reaction $\alpha$, such that $\sum_{l=1}^{\alpha-1} r_l < p_2 r_{tot} \leq \sum_{l=\alpha}^{m_1} r_l$
   
   - Update species number $x = x + \nu_{\alpha}$

10. else

    - Set time step $\Delta t = \Delta h t$ and update $t = t + \Delta t$

11. Generate $m_2$ random numbers $R \sim N(0,1)$

12. Update the Langevin part $x_h = x_h + r_h \Delta t + \sqrt{r_h \Delta t} \cdot R$
13. Repeat steps 2-11, while $t < t_{\text{final}}$

This algorithm is much faster than the full SSA when the reaction network has a “fast” reaction subset and it still captures fluctuations of reactions with high molecule levels via Langevin equations. The separation of reactions is done inside the algorithm and not off-line. This gives the system the flexibility to run stochastically for reactions with low molecule numbers and via Langevin equations for reactions with high molecule levels. It can also be extended for use with a DSSA. The same algorithm can be used for the initiation reactions of the DSSA, but in addition $\Delta t$ and $\Delta h t$ have to be compared to the first stored completion time of the delayed reactions $t_d = \min(T_d)$. If $t_d \leq t + \Delta t$ and $t_d \leq t + \Delta h t$, the delayed reaction will be completed. If not, steps 9-11 of the hybrid Langevin algorithm have to be followed and if a delayed reaction is initiated, store that at time $t + \tau$ the system must be updated according to the completion of the delayed reaction.

The QSSA can be used for the fast switching state in the system. If a state equilibrates to its steady-state condition on a time scale fast compared to the evolution of other species that are involved in reactions with this state, then we find the steady state probability density for this state as mentioned earlier. In the case of the $N$ protein, the simulation switches to the QSSA of $N$ at around $t \approx 4000 s$. Instead of modeling all production reactions, the $N$ protein level is calculated by drawing randomly from its probability density at each time step. The approximated $N$ protein level is used to calculate the initiation reaction rate $r_{\text{encap}}$ and the delay time $\tau_{\text{encap}}$ of each encapsidation reaction. When the mean $\alpha$ of the $N$ steady-state probability density is changing, all stored reaction delays $T_{d,\text{rep}}$ have to be updated using the new delay time.
\[
\begin{align*}
    r_{\text{encap}}(i) &= k_3 \cdot \alpha(i) \\
    \tau_{\text{encap}}(i) &= 1258 \cdot \frac{1}{\tau_{\text{encap}}(i)} \\
    T_{d,\text{rep}}(i) &= \frac{T_{d,\text{rep}}(i - 1) - \Delta}{\tau_{\text{chain}}(i - 1)} \cdot \tau_{\text{chain}}(i)
\end{align*}
\]  

(4.1)

Although the QSSA on \( N \) allows us to take bigger time steps, the \( L \) protein also switches fast at low molecule levels and slows down the simulation. Therefore it is still not possible to simulate to the end of the viral infection because an approximation for the delayed species \( L \) has yet to be developed.
Table 4.1: List of mathematical letters, symbols, and abbreviations. Listed in order of appearance.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n_{sw}$</td>
<td>Switch to separate “high” and “low” molecule number reactions</td>
</tr>
<tr>
<td>$m_1$</td>
<td>Number of low molecule reactions</td>
</tr>
<tr>
<td>$m_2$</td>
<td>Number of high molecule reactions</td>
</tr>
<tr>
<td>$tol$</td>
<td>Tolerance for the maximum change in Langevin species</td>
</tr>
<tr>
<td>$t$</td>
<td>Simulation time</td>
</tr>
<tr>
<td>$x$</td>
<td>Vector of species numbers</td>
</tr>
<tr>
<td>$x_0$</td>
<td>Initial condition of species numbers</td>
</tr>
<tr>
<td>$r_j$</td>
<td>Reaction rate of reaction j</td>
</tr>
<tr>
<td>$k_j$</td>
<td>Rate constant of reaction j</td>
</tr>
<tr>
<td>$k_j a_j(x)$</td>
<td>Reaction rate of reaction j</td>
</tr>
<tr>
<td>$r_{tot}$</td>
<td>Total reaction rate</td>
</tr>
<tr>
<td>$p$</td>
<td>Uniformly distributed random number on (0,1)</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>Stochastic time step</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>Maximum Langevin time step</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>Next time step</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Stochiometric matrix</td>
</tr>
<tr>
<td>$n$</td>
<td>Next reaction or row of stochiometric matrix $\nu$</td>
</tr>
<tr>
<td>$R$</td>
<td>Normally distributed random number with $\mu = 0$ and $\sigma = 1$</td>
</tr>
<tr>
<td>$t_{final}$</td>
<td>Length of simulation</td>
</tr>
<tr>
<td>$t_d$</td>
<td>Earliest stored completion time of delayed reactions</td>
</tr>
<tr>
<td>$T_d$</td>
<td>Vector of all stored delayed reaction completion times</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Reaction delay</td>
</tr>
<tr>
<td>$r_{encap}$</td>
<td>Initiation rate of delayed encapsidation reaction</td>
</tr>
<tr>
<td>$\tau_{encap}$</td>
<td>Delay of encapsidation reaction</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Mean of the QSSA probability density of the $N$ protein</td>
</tr>
<tr>
<td>$T_{d,rep}$</td>
<td>Vector of all stored encapsidation reaction completion times</td>
</tr>
</tbody>
</table>
Chapter 5

Results and Discussion

The complexity of the reaction network lies in the nature of the fast switching species and the delayed reactions. We are able to implement our model using Langevin equations and the QSSA assumption for the $N$ protein, which allows us to take bigger time steps and simulate further in time. Fig. 5.1, Fig. 5.2 and Fig. 5.3 show the means of the genomes, the viral mRNAs and the viral proteins, respectively. In Fig. 5.1 we find that the first (+)RNA appears shortly after one hour, which is the time it takes to encapsidate and replicate the (-)RNA genome. Due to the delay and the attenuation at each gene junction, our simulation shows a gradual decrease of the mRNA level (Fig. 5.2 in the order of $N > P > M > G > GFP > L$). The proteins in Fig. 5.3 show the same order for the $P$, $M$, $G$ and $GFP$ proteins. These proteins are not involved in other reactions. The $N$ and the $L$ proteins are both fast switching species. They are produced and consumed and in the case of the $L$ protein also released. As we are only tracking free protein levels, these two proteins do not follow the same order. We can also see that these two proteins still fluctuate, because they are both highly reactive species, and we would need more simulation runs to get a better mean.

We also perform 1,000 simulations of the infection with a recombinant GFP-N VSV virus. The reaction network has to be slightly changed in order to put the GFP into the first position instead of the fourth. Fig. 5.4 compares the mean of the GFP level of both models to the GFP level in our measurements. In order to compare the GFP level with experimental data, the simulation results have been shifted about two hours (128.4 mins) to the right, because that is the amount of time it takes from the formation of the GFP protein until fluorescence is emitted, [19]. In our experimental work, we only measured intensity of the GFP protein and not its actual molecule level. A single adjustable parameter to scale the
Another more intriguing result is the new phenomena of clustering in the population of (-)RNA genomes. These separations of population occur because of the sensitivity to the lowest number of viral species in the cell, the L protein. In this work, the simulation runs are stopped at $t = 4$ hours. We first perform a simulation of the G-GFP recombinant virus and look at the distributions of several viral constituents at different points in time. Fig. 5.5 shows the distribution of the (-)RNA genome level for 1,000 simulation runs of the intensity measurements is used to fit each curve to its data. We get a reasonable match and can see that the GFP protein in the G-GFP strain shows up much later than in the GFP-N strain, for both the simulations and the measurements. The difference between the two strains is only the position of the GFP gene in the genome. We were expecting this result because the GFP gene in the GFP-N strain (and model) is at the first position, and therefore is expressed earlier and with a higher rate compared to the fourth position of the GFP gene in the G-GFP strain (and model).

Figure 5.1: This plot shows the mean of the genome levels over time. (-)RNA includes all naked, partially and fully encapsidated (-)RNA strands and (+)RNA includes all naked, partially and fully encapsidated (+)RNA strands.
Figure 5.2: This plot shows the mean of all mRNA levels over time.

Figure 5.3: This plot shows the mean of all protein levels over time.
Figure 5.4: This plot compares the mean of the GFP molecule level of both models to the mean of our experimental results. The GFP-N model (solid lines) and experimental (▲) molecule levels are expressed earlier and with higher rates than the GFP level in the G-GFP model (dashed lines) and experimental results (■).

G-GFP recombinant virus. The (-)RNA genome level includes all naked, partially and fully encapsidated (-)RNA strands. We can see that the mean is increasing over time and that the distribution shows separation of populations beginning at about 2.5 hours post infection (HPI). This is due to the fact that the event of forming L mRNA is so rare that there are simulation runs that lead to a lower or higher distribution depending on how many L mRNAs have been expressed in the first round of transcription. These different amounts of L mRNA from the first transcription of the (-)RNA genome lead to the clustering in total (-)RNA genome production. The amount of L mRNA will stay constant until the first L mRNA strand is expressed from the first replicated (-)RNA strands. The presence of L protein is necessary for gene expression and genome replication, and the earlier the L mRNA, and therefore L protein is formed, the faster the virus can produce all other species. Fig. 5.6 shows how dependent the separation of (-)RNA genome population is to the level
Figure 5.5: This plot shows the distributions of the (-)RNA genome level for 1,000 simulation runs of the G-GFP recombinant virus. The (-)RNA genome level includes all naked, partially and fully encapsidated (-)RNA strands. The numbers above the different populations indicate the corresponding number of L mRNA molecules in the system at $t_0 = 1.5$ hours.
Figure 5.6: This plot shows the dependence between the (−)RNA genome level at \( t = 4 \) hours and the \( L \) mRNA level at \( t_0 = 1.5 \) hours. The numbers indicate the amount of simulations that expressed the corresponding level of \( L \) mRNA at \( t_0 = 1.5 \) hours.

of \( L \) mRNAs. The level stays constant until the second round of replication, which does not happen before 3 HPI. The \( L \) mRNA level is measured at \( t = 1.5 \) hours, and the higher the level is at that point in time, the more (−)RNA genomes are produced at \( t = 4 \) hours.

Fig. 5.7 shows the GFP molecule distribution at different HPI. The GFP distribution shows a similar pattern as compared to the distribution of the (−)RNA genomes. Both show a separation of populations, but the separation of the GFP protein level occurs later, due to the delayed impact (transcriptional delay) of the (−)RNA genomes on the mRNAs. Recent stochastic models have shown bifurcations in population. The bifurcation in the HIV-1 phenotypes suggested the implementation of a Tat feedback loop. The model results of that work successfully predicted experimental results [10]. Studies of the λ phage have shown a lysis-lysogeny decision circuit and a model by Arkin [11] shows how stochastic fluctuations influence the life cycle decision by implementing a threshold mechanism for different promoter activations. Srivastava’s generic model [12] has also shown that the stochastic solution of
Figure 5.7: This plot shows the distributions of the GFP protein level for 1,000 simulation runs of the G-GFP recombinant virus. The numbers above the different populations indicate the corresponding number of $L$ mRNA molecules in the system at $t_0 = 1.5$ hours.
a generic virus model leads to two different steady states of the system, features that could not be explained by the deterministic model. The reason for the separation of distribution in this work is a little bit more subtle. Modeling the important mechanisms in the reaction network shown in Fig. 1.1 led to the separation of population. It is due to the different level of mRNA of the virus’ least abundant species, the $L$ protein, which was produced from the first (-)RNA strand before it was converted into its fully encapsidated form. Differences in integer values of the $L$ mRNA at early times in an infection cycle lead to different populations in the (-)RNA genome level which can be seen in Fig. 5.6.

Different levels of viral constituents from different virus strains have been examined in a previous deterministic model, but the model was not capable of producing the separation of population or distribution of species as we have seen it in this stochastic model. The goal of this work was to model a viral infection of the VSV, and to gain insights on which reactions most prominently lead to the distributions of the viral constituents and the virus yield. We were able to show a reasonable match for the GFP molecule level of the GFP-N and G-GFP models compared to experimental measurements. Another new result was the phenomena of separation of population that occured because of the different amount of $L$ mRNA transcribed from the first (-)RNA strand. The stochastic methods that have been developed in the past decades are not yet capable of simulating the full viral infection cycle for the reaction network given in this work.
Chapter 6

Conclusions

6.1 Contributions

The contributions to research from this work are not only the simulation results and the comparison to experimental data, but also the successful implementation of stochastic simulation methods to overcome the problem of small time steps. Besides underlining experimental results and expected rankings of viral constituents, we discovered a new phenomena. The separation of population of several viral constituents is a feature that has not been accounted for during the implementing of the reaction network. The Langevin equations that have been included in the DSSA already help to take bigger time steps. The QSSA on the fast switching species also provided us with a reduced model description that allowed for faster simulations. However, there is no method yet to approximate the fast switching species that are involved in the delayed reactions accurately enough. Therefore the step size of the simulation will be dictated by the consumption and the release of these delayed reactions. Although recent methods have focused on reducing the complexity and the computational burden of stochastic reaction networks that involve fast reactions or highly reactive species, these methods do not aid in completing the simulation to the end of the viral infection model examined in this work. However, we were still able to simulate the model up to an earlier stage in the infection cycle. The effect of the produced $L$ mRNA level early on seems to be the key factor in the outcome of a viral infection and contributes significantly to the distribution and population of all other viral species. Infections by different viruses generally involve rapid amplification of small numbers of few components, as it is shown by the impact of the $L$ mRNA in this model, we expect the Langevin and the QSSA approach on the fast
switching species to find broad application to the stochastic simulation of diverse viruses. These simulations could lead to better understanding of the influence of viral mechanisms at early stages of an infection cycle and therefore provide more insight into the design of anti-viral therapies.

6.2 Future Work

We used several simulation techniques in our model to overcome the problem of small time steps. Using the QSSA on the $N$ protein and Langevin equations for the “high” molecule reactions allowed us to take bigger time steps and to simulate much further in time. Although we were able to match our model to experimental data and see the separation of population as an intriguing new phenomena, some methods have to be refined in order to simulate further in time and to create a better approximation for the full stochastic model. Switching to a different implementation to approximate all 1,258 chain reactions for the encapsidation process is unavoidable, but we might have lost some stochastic effects by implementing all reactions as one delayed reaction. How much we differ cannot be tested because we do not have a full stochastic solution that we could compare it to and future work has to be done to solve these kinds of problems. In a simpler QSSA model in B, it is shown that the chain reactions can be approximated by using only the mean of the probability distribution of the fast switching species, but further analysis has to be done to better understand the problem.

The reason we were not able to simulate any further is because of the fast switching $L$ protein that is involved in the delayed reactions. More analysis has to be done to calculate a probability density, as we calculated it for the $N$ protein. It is a hard problem because of the effect of the delayed reactions initiated in the past that influence the level of $L$ in the present. When this distribution can be calculated, the system could be updated via delayed
Langevin equations for the species at high molecule levels that are involved in the delayed reactions.

Solving these problems should help to simulate a full virus infection. These simulation results can then be compared to more experimental data, which would help to better estimate uncertain parameters. Most of the parameters we used in this model have been derived from experimental measurements, while others have been estimated in our previous deterministic model, such as the ribosomal spacing.

Another challenge is the complexity of the sensitivity analysis on stochastic models. It would help to gain insight into the processes that contribute most prominently to the observed distributions, but while there are many methods to analyze the parameter sensitivity of deterministic models, methods to estimate complex stochastic networks have yet to be developed.

In our model, translation reactions have been implemented without delays. One reason is that there are much shorter delays compared to the transcription reactions and another reason is the uncertainty in the ribosomal spacing. Assuming abundance of host ribosomes, the ribosomal spacing defines the rate of the protein production and delays could be neglected. Therefore, we have to ask the question if translation reactions should ever be implemented as delayed reactions.

The clustering in the population of several viral constituents has not yet been measured experimentally and future work has to be done to match the time of absorption. This is necessary because the time of infection has to be synchronized in order to compare it to the simulation results.
Bibliography


Appendix A

Ribosome Footprint

The ribosome footprint is a parameter that we obtained from our previous deterministic model because there are no exact measurements of the footprint in the literature. The ribosome footprint, combined with the ribosome elongation rate, let us calculate the translation rate of all viral proteins. Therefore, this single parameter has a big influence, not only on the proteins, but also on all other species that are influenced by the protein molecule levels. The bigger the footprint, the fewer the number of ribosomes that can translate the mRNAs, which leads to a lower production rate of the proteins. This can be seen in table A.1, which shows how the ribosomal footprint influences the mean of the $N$ protein level at $t = 4$ hours. A lower mean for the $N$ protein leads to a longer delay for the encapsidation reactions, which lowers the autocatalytic feedback of the (+)RNA and (-)RNA genomes.

<table>
<thead>
<tr>
<th>Ribosome footprint $s_{rib}$</th>
<th>Mean of $N$ protein level $\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 nt $s_{rib} = 200$ nt</td>
<td>$\alpha = 5.887$</td>
</tr>
<tr>
<td>238.5 nt $s_{rib} = 238.5$ nt$^1$</td>
<td>$\alpha = 4.598$</td>
</tr>
<tr>
<td>400 nt $s_{rib} = 400$ nt</td>
<td>$\alpha = 3.032$</td>
</tr>
<tr>
<td>600 nt $s_{rib} = 600$ nt</td>
<td>$\alpha = 2.323$</td>
</tr>
<tr>
<td>800 nt $s_{rib} = 800$ nt</td>
<td>$\alpha = 1.457$</td>
</tr>
<tr>
<td>1,000 nt $s_{rib} = 1,000$ nt</td>
<td>$\alpha = 0.316$</td>
</tr>
</tbody>
</table>

$^1$s_{rib} = 238.5 nt is the estimated value of our previous deterministic model and the value we are using for our simulations.
Appendix B

QSSA Example Model

The following example reaction network is used to test the method of the QSSA that we used to calculate the underlying distribution of the fast switching N protein. It is a simpler version with only 100 chain reactions, but it captures the main features of the full virus model.

\[
\begin{align*}
A_0 + B & \xrightarrow{k_1} A_1 \\
A_1 + B & \xrightarrow{k_1} A_2 \\
\vdots & \xrightarrow{k_1} \vdots \\
A_{99} + B & \xrightarrow{k_1} A_{100} \\
C + S_1 & \xrightarrow{k_2} C + B
\end{align*}
\] (B.1)

The parameters for this model are given in B.1. \(S_1\) is an unlimited resource and does not affect the reaction rates that are given in B.2. \(A_{100}\) is the final product and does not react with anything.

\[
\begin{align*}
r_{1(1)} &= k_1 A_0 \cdot B \\
r_{1(2)} &= k_1 A_1 \cdot B \\
\vdots &= \vdots \\
r_{1(100)} &= k_1 A_{99} \cdot B \\
r_2 &= k_2 C
\end{align*}
\] (B.2)
Table B.1: Model parameters for the QSSA example reaction network.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chain reaction rate constant</td>
<td>$k_1$</td>
<td>0.0461 $\text{sec}^{-1}$</td>
</tr>
<tr>
<td>$B$ production reaction rate constant</td>
<td>$k_2$</td>
<td>0.0461 $\text{sec}^{-1}$</td>
</tr>
<tr>
<td>Initial condition</td>
<td>$A_0$</td>
<td>100</td>
</tr>
<tr>
<td>Initial condition</td>
<td>$A_1$ to $A_{100}$</td>
<td>0</td>
</tr>
<tr>
<td>Initial condition</td>
<td>$B_0$</td>
<td>0</td>
</tr>
<tr>
<td>Initial condition</td>
<td>$C_0$</td>
<td>$1 \times 10^4$</td>
</tr>
</tbody>
</table>

$B$ is a highly reactive state in the system. It is switching between low values such that they cannot be approximated by the Langevin equations. $B$ is produced by $C$ with a high rate, while it is consumed by all $A_i$ except for $A_{100}$. In the following, $\tilde{A}$ stands for the sum of all $A_i$ except for $A_{100}$. All rates for the chain reactions are calculated by the same reaction rate constant $k_1$. $C$ is present in large amounts and the sum of all $A$ is constant most of the time. Considering the sum of $A$ to be constant over some interval of interest, we can derive the following master equation for $B$:

$$\frac{dP(B,t)}{dt} = -k_2CP_B + k_2CP_{B-1} - k_1\tilde{A}BP_B + k_1\tilde{A}(B+1)P_{B+1}$$  \hspace{1cm} (B.3)

$P_B$ is shorthand for $P(B,t)$. The production and consumption rates are high, such that $B$ equilibrates to its steady-state condition on a fast time scale compared to the evolution of level $A_i$ and $C$, the steady-state probability density of $B$ can be found by setting $dP(B,t)/dt = 0$. Using

$$r_1 = \tilde{A}$$

$$r_2 = k_2C$$  \hspace{1cm} (B.4)
the following equation can be derived:

\[ 0 = -r_2 P_B + r_2 P_{B-1} - r_1 B P_B + r_1 (B + 1) P_{B+1} \]  
(B.5)

Evaluating this equation for \( B = 0, 1, 2, \ldots \)

\[
\begin{align*}
B = 0 & : 0 = -r_2 P_0 + r_1 P_1 \\
B = 1 & : 0 = -r_2 P_1 + r_2 P_0 - r_1 P_1 + 2r_1 P_2 \\
B = 2 & : 0 = -r_2 P_2 + r_2 P_1 - 2r_1 P_3 + 3r_1 P_3 \\
\vdots & : \vdots \\
B = n & : 0 = -r_2 P_n + r_2 P_{n-1} - n r_1 P_n + (n + 1) r_1 P_{n+1} \\
\end{align*}
(B.6)

This relation provides the following recursion in terms of \( P_0 \).

\[
\begin{align*}
P_B & = \frac{1}{B!} \alpha^B P_{B-1} \\
P_B & = \frac{1}{B!} \alpha^B P_0, \quad \alpha = \frac{r_2}{r_1} = \frac{k_2 C}{k_1 A} \\
\end{align*}
(B.7)

Summing \( P_B \) over \( B \) gives us:

\[
\sum_{B=0}^{\infty} P_B = (1 + \frac{\alpha}{1!} + \frac{\alpha^2}{2!} + \frac{\alpha^3}{3!} + \cdots) P_0 \\
1 = e^\alpha P_0 \\
P_0 = e^{-\alpha}
\]  
(B.8)

The quasi-steady probability density of \( B \) is therefore:
\[ P_B = \frac{1}{B!} \alpha^B e^{-\alpha} \]  

(B.9)

Fig. B.1 shows three full model simulation runs and Fig. B.2 shows three QSSA simulation runs using only the mean \( \alpha \) of the QSSA distribution on \( B \) to calculate the chain reaction rates. We can see that the occurrences of \( A_{50} \) and \( A_{100} \) happen around the same time, and therefore, the simulation results are similar. Thus, the mean is the only parameter that we need from the QSSA distribution of \( B \) to calculate the influence on all other species. The fluctuation of \( B \) is uncoupled from the fluctuation of arrival times of \( A_{100} \), because the delay is large compared to the time scale of the fast reactions involving the highly reactive species \( B \). The calculated distribution could be used to sample the \( B \) molecule level at different points in time. Fig. B.3 shows the QSSA distribution for three different values of \( k_1 = 10 \cdot 0.0461 \), \( k_1 = 0.0461 \), and \( k_1 = \frac{1}{10} \cdot 0.0461 \), respectively. We can see that the full model probability distribution of 100 full model simulation runs matches nicely the calculated QSSA probability distribution of \( B \).

Using this approximation, we can calculate the QSSA distribution of the \( N \) protein level in the full model and use only the mean \( \alpha \) of the underlying distribution to calculate the delay of all chain reactions.
Figure B.1: This plot shows the molecule levels of $A_0$, $A_{50}$ and $A_{100}$ for three different simulation runs of the full stochastic model.

Figure B.2: This plot shows the molecule levels of $A_0$, $A_{50}$ and $A_{100}$ for three different simulation runs of the QSSA model using only the mean $\alpha$ of the QSSA distribution on $B$ to calculate the chain reaction rates.
Figure B.3: This plot compares the distribution of the $B$ molecule level derived from full model simulations to the calculated QSSA probability density of $B$ for three different parameters $k_1 = 10 \cdot 0.0461$, $k_1 = 0.0461$, and $k_1 = 1/10 \cdot 0.0461$, respectively.