Systems Approach to Microbial Pathogenesis: Complex Patterns Emerge from Simple Interactions

Suzy M. Vasa

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree Doctor of Philosophy in the Department of Biomedical Engineering.

Chapel Hill
2009

Approved by:

Morgan C. Giddings

Shawn M. Gomez

Oleg V. Favorov

Jennifer Webster-Cyriaque

Thomas M. O'Connell

© 2009
Suzy M. Vasa
ALL RIGHTS RESERVED

Abstract

SUZY M. VASA: Systems Approach to Microbial Pathogenesis: Complex Patterns Emerge from Simple Interactions
(Under the direction of Morgan C. Giddings)

Biological organisms are complex systems and modeling can provide insight into their behavior by the process of recreating it. All elements may not be known of the system under study and thus, hypotheses must be made in order to create an appropriate model. These hypotheses can lead to interesting modeling results and help guide *in vitro* experiments. However, modeling complexity does not necessarily require complex techniques. By modeling the simplest elements of a biological system and by defining how the elements interact, it is possible to model complex behavior as emergent properties of the system. In this manner, I model simple interactions between biological elements. First, at the lowest level of complexity, is a single molecule such as an RNA. Determining RNA secondary structure is a necessary step to understand how it interacts with other molecules to affect the biological system as a whole. The structure of an RNA is formed through simple interactions between nucleotides. I developed software that aids the process of identifying sites in an RNA where nucleotide-nucleotide or nucleotide-protein binding occurs to predict RNA secondary structure more accurately. The next level of complexity is molecule-molecule interactions that result in the emergence of patterns within an organism,

such as phenotypes expressed by a cell. Using agent-based modeling, I model the proteins, RNAs, and enzymes involved in a gene regulatory network that is responsible for the emergence of the competence phenotype in *Bacillus subtilis.* Competence is stochastically expressed due to the variable expression of genes. My agent-based model identified several possible sources for this variation: dilution events like cell division, inheritance of molecules involved in competence and most importantly, spatial temporal interactions of molecules. And lastly, I model the simple interactions between two organisms, a virus and a host cell, to understand the molecular interactions between host and pathogen that result in the replication and assembly of a virus. In this model, I successfully modeled the self-assembly of BK Virus using an agent-based model that models from transcription to translation to the encapsidation of the BKV genome within a T=7, icosahedral structure all by simple molecule-molecule interactions.

Acknowledgements

I wish to thank the members of my committee, the Giddings, Webster-Cyriaque and Weeks Lab for graciously providing their knowledge and support. I also wish to thank my children, Aaron and Asher, for being so patient with me.

A portion of this dissertation was supported by F31DE019594 from the National Institute of Dental & Craniofacial Research. The content is solely the responsibility of myself and does not necessarily represent the official views of the National Institute of Dental & Craniofacial Research, the National Center for Research Resources, or the National Institutes of Health.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ABM – Agent-based model

BKV – BK Virus

CER – Cytoplasm/Endoplasmic Reticulum

DNA – Deoxyribonucleic Acid

HIV – Human Immunodeficiency Virus

ODE – Ordinary Differential Equations

RNA – Ribonucleic acid

SGD – Salivary Gland Disease

SHAPE – Selective 2’-Hydroxyl Acylation by Primer Extension

VLP – Virus Like Particle

Chapter 1
Introduction

A biological organism is a complex system of multiple interacting molecular pathways comprised of numerous biochemical interactions. Self-organization or complexity in biology occurs through seemingly simple biochemical interactions that give rise to complex patterns and phenotypes such as stripes on a zebra, bacterial phenotypes or viral capsid assembly [1, 2]. These not easily predicted biological patterns manifest over time from seemingly simple interactions, Figure 1.1. Modeling the underlying complexity of these biological patterns remains a challenge and can be quite daunting. Approaches to dissect the biology of a cell range from the study of a specific molecule to the study of gene and protein interaction networks in an attempt to break such a large, complex problem into more manageable, smaller pieces. Computational modeling approaches such as top-down methods like mathematical modeling tend to take the latter approach and tackle the problem by attempting to identify patterns, motifs or modules in a biological system and model these components [3].

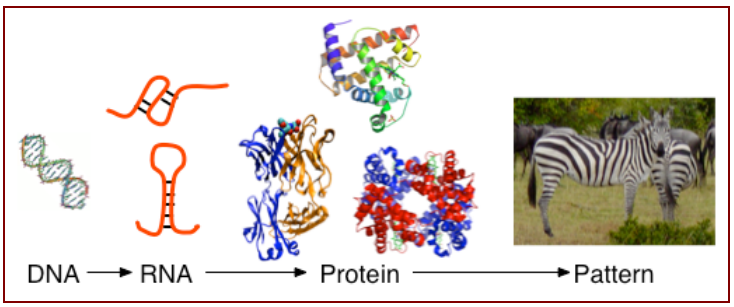

Figure 1.1. Simple interactions between molecules lead to the expression of different phenotypes.

Instead of modeling individual components, my work attempts to find a balance between simplification and complexity of biological systems by modeling the simple biochemical interactions. These simple interactions result in the emergence of a global phenotype or complex structures. First, simple interactions between nucleotide bases of the HIV-1 genome were studied to solve its secondary structure. Second, simple interactions involved in a gene regulatory network were modeled to unravel the sources of variability in the expression of the competence phenotype of the gram-positive bacteria *Bacillus subtilis*. Lastly, the interaction of host cell machinery with the viral replication and assembly process of BK virus was studied to comprehend the pathogenesis of BKV within salivary gland cells.

1.1 Simple Interactions of a Single Molecule, HIV-1 Genome

Most RNAs perform their biological function(s) only after they fold to form two and three-dimensional structures, Figure 1.1. As an RNA forms a preferred secondary or tertiary structure, a subset of nucleotides becomes conformationally constrained by the simple interaction of bases pairing and tertiary interactions, while other nucleotides are left unpaired and unconstrained. The HIV genome consists of a single stranded RNA molecule consisting of 9 genes coding for 19 proteins [4]. The

function of the HIV genome is tightly linked to its structure in terms of conformational changes during the progression of infection while interacting with transcription factors, replication complexes and structural proteins during transcription, replication and packaging [5]. For instance, the structure of the primer binding site (PBS) as seen in Figure 3.1 undergoes a conformation change when the tRNA primer binds to the large loop region [6]. This binding stabilizes the tRNA primer, which then is incorporated into viral particles and is necessary for the initiation of reverse transcription [4].

I developed a software system to aid in secondary structure prediction of an RNA in order to help identify biological function of domains of the HIV genome as described in Chapters 2-4. The software takes a chromatogram of a Selective 2'-Hydroxyl Acylation analyzed by Primer Extension (SHAPE) experiment [6-8]. The cDNA products from each reaction are combined and separated on an automated capillary electrophoresis instrument of the type commonly used for high-throughput DNA sequencing. A single SHAPE experiment measures backbone flexibility of more than 300 RNA nucleotides at a time; multiple experiments can be combined for the analysis of RNAs of any length. The quantified results can then be used as input to third party secondary structure prediction algorithms.

SHAPE provides valuable information about local backbone flexibility, but quantifying the per nucleotide flexibility information is a difficult, time-consuming task. *rnafit*, a software tool to aid this process, is a command line application that processes SHAPE experiments by aligning sequencing peaks with the RNA sequence and calculating peak areas to quantify per-nucleotide flexibility. Figure 1.2

shows example output from *rnafit*. Previously, users would search through this output to determine peak finding accuracy and sequence alignment accuracy. To provide a more user-friendly, graphical user interface (GUI) and to provide a more integrated signal-processing platform, I integrated *rnafit* into BaseFinder and created additional signal processing tools. BaseFinder is a software system originally designed to analyze spectral data output from DNA sequencing equipment [9]. It is based on an extensible, modular software architecture that easily allows the addition of new analysis algorithms in the form of "tools".

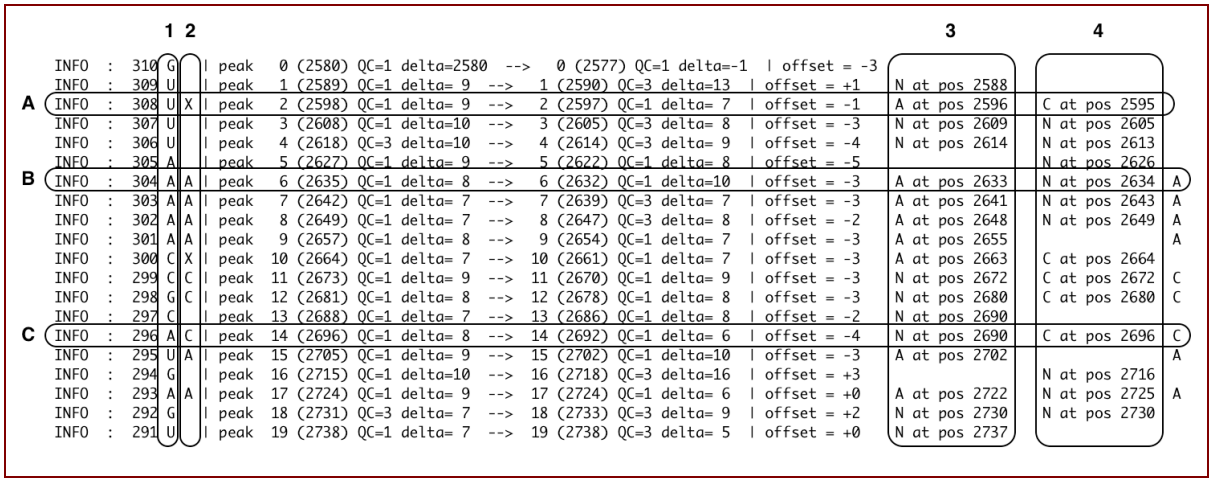

Figure 1.2 *rnafit* version 0.82 output. Column 1 is the RNA sequence. Column 2 is the aligned sequence. Columns 3 and 4 give feedback on the alignment of the sequencing lanes. A), B) and C) give examples of alignment and misalignment. An X indicates the algorithm was uncertain of the nucleotide in the alignment.

I created a new tool called *Align and Integrate* that incorporated the *rnafit* algorithm, described in Chapter 2. In addition, I performed the statistical analysis as well as create the new signal processing tools of *Scale Factor*, *Mobility Shift: Cubic* and *Signal Decay Correction,* also described in Chapter 2, resulting in the new software platform ShapeFinder based on BaseFinder [10].

Chapter 3 details improvements I developed for the sequence alignment algorithm of the original *rnafit* software integrated within the *Align and Integrate* tool.

Lastly, there are two published reagents which bind to the 2'-OH of a nucleotide used in SHAPE chemistry: N-methylisatioic anhydride (NMIA) and 1-methyl-7-nitroisatoic anhydride (1M7) [7, 11]. However, a detailed statistical analysis was needed to determine whether or not the reagents exhibit sensitivity to base identity. In other words, do the reagents react equally independent of nucleotide type? Thus, a series of experiments were performed to obtain denatured SHAPE reactivity data of four different RNAs: 976 nts from the 5' end of the HIV-1 genome, the 154 nt specificity domain of *Bacillus subtilis* RNase P, and ~400 nt internal segments of the *Escherichia coli* 16S and 23S rRNAs. Both NMIA and 1M7 data was obtained for each RNA and the statistical analysis was performed using a Bootstrap ANOVA detailed in Chapter 4 [12].

Bootstrapping [13, 14] is based on the theory that even though the distribution of the population our data was collected from is unknown, the empirical distribution is a close approximation. Essentially, I estimated the true distribution by repeated sampling from the empirical distribution of the data set. It is a computationally expensive, yet distribution free technique I used to simulate repeated SHAPE experiments to determine if the differences seen between the groups of nucleotides were due to the reagent or were due merely to chance. Given that the SHAPE measurements are independent and by using the pivotal F-statistic, the bootstrap procedure is robust for SHAPE data.

1.2 Simple Interactions of a Gene Regulatory Network, Competence in *B. subtilis*

The study of a single molecule does not provide significant insight into the complex interactions involved in gene regulatory networks, but is useful to identify function and interaction partners. Many proteins, RNAs and enzymes interacting with one another lead to the emergence of phenotypes and patterns in nature, Figure 1.1. Following this principle, I modeled a gene regulatory network in order to understand the stochastic nature of the emergence of differing phenotypes in genetically identical bacteria cells. I selected to model the competence gene regulatory network of *Bacillus subtilis* as described in Chapter 5 as it is a well-studied phenomenon.

Genetically identical bacterial cell populations can express various phenotypes due to stochastic events and environmental input [15]. There is a growing body of evidence demonstrating that transitions from one bacterial cell phenotype to another are often governed by regulatory feedback loops [16]. This is called bi-stable switching where we have a system with two states, enabled or disabled.

In a *B. Subtilis* cell, a bi-stable switch controls the competence phenotype that enables the uptake of DNA from the environment. Approximately, 10-20% of a *B. subtilis* population will express the competence phenotype [17]. The metabolic pathway that enables the competence switch is controlled by cell density and nutritional status and it is highly regulated [17]. It has been shown that the random expression of the competence phenotype is due to the variable expression of the *comK* gene [18]. ComK acts as its own transcription factor and thus positively auto-regulates itself. ComK is the “switch” that enables the competence switch.

Identifying the mechanisms that lead to the random expression of the *comK* gene is difficult to identify experimentally and the agent-based model (ABM) I created helps visualize this process in order to understand the operation of this system.

In Chapter 5, I describe an ABM of the metabolic pathways that control the competence switch in *B. subtilis* to study the simple interactions of proteins and other molecules that comprise a gene regulatory network. At the lowest level, my *B. subtilis* cell ABM, agents represent the proteins and regulatory elements of the competence metabolic pathway. A virtual 3-D environment is created where an agent diffuses throughout the cell landscape and is influenced by random interactions with other agents. When an agent "bumps" into another agent, rules for interaction between agents are exercised when applicable, such as dimerization, protein binding, transcription, translation, protein or mRNA degradation, etc. The ABM easily models the stochastic events in the molecular pathway, thus, allowing observations on the emergence of competence phenotypic behavior.

At the next level, colony growth by cell division is then modeled by considering each "Cell ABM" as an agent. Changing nutrient levels in the model and colony growth influence each cell agent's metabolic pathways such that the competence state randomly emerges in a subpopulation of the *B. subtilis* colony ABM. The model demonstrates the transition of 10-20% of the *B. subtilis* colony ABM to the competence phenotype and highlights three interesting observations: (i) spatial temporal interactions control the competence switch, (ii) molecules inherited by daughters cells influence the emergence of competence and (iii) cell division regulates competence emergence.

1.3 Simple Interactions of Interacting Organisms, Virus-Host Cell

Finally, modeling interacting biological networks is another method of understanding how complex patterns, motifs or modules arise from simple biochemical interactions. In the case of virus-host cell interactions, I describe in Chapter 6 a model of the interaction between two organisms to study the emergence of disease.

It is well established that viruses have cell transforming properties and can induce tumor formation and diseases [4]. One such virus is the BK Virus (BKV). BKV, a polyomavirus family member, is a non-enveloped, small, double-stranded DNA virus. BKV is believed to cause a harmless latent infection in healthy people but may reactivate if the immune system has been compromised [19]. Recently, BKV has been detected in HIV positive patients with HIV associated salivary gland disease (HIV SGD) and shown capable of reproducing in salivary gland cells [20]. As salivary gland diseases such as HIV SGD or Sjögren's Syndrome do not have a known etiological agent, I developed a computational model to pursue this relationship with BKV.

To understand further the association of BKV with salivary gland disease, I modeled BKV replication in a salivary gland cell in order to create simulations of viral pathogenesis. I created a spatial-temporal molecular model using agent-based modeling (ABM). Agents were used to model the synthesis of the capsid proteins and the BKV genome interacting with host molecular agents. Additionally, agents via interactions by simple rules of attraction, repulsion and movement form complete capsid bound infectious and non-infectious BKV particles.

This model focused on the synthesis of the capsid proteins and self-assembly of the BKV virion to capture viral rates of production. Host cells essentially determine the growth rate of a virus, and this model mimics the salivary gland cell support and hindrance of the BKV life cycle. There is so little understood about the replication cycle of BKV and it has just recently been associated with SGD [20]. Eventual additions to this model of the complete BKV replication cycle will yield insights into the pathogenesis of this disease. Eventually, the goal is to create a model of the organ leading to a model of the infection of the human host.

Chapter 2
ShapeFinder: A software system for high-throughput quantitative analysis of nucleic acid reactivity information resolved by capillary electorphoresis

2.1 ABSTRACT[1]

Analysis of the long-range architecture of RNA is a challenging experimental and computational problem. Local nucleotide flexibility, which directly reports underlying base pairing and tertiary interactions in an RNA, can be comprehensively assessed at single nucleotide resolution using high-throughput selective 2'-hydroxyl acylation analyzed by primer extension (hSHAPE). hSHAPE resolves structure-sensitive chemical modification information by high-resolution capillary electrophoresis and typically yields quantitative nucleotide flexibility information for 300-600 nts per experiment. The electropherograms generated in hSHAPE experiments provide a wealth of structural information; however, significant algorithmic analysis steps are required to generate quantitative and interpretable data. We have developed a set of software tools called ShapeFinder to make possible rapid analysis of raw sequencer data from hSHAPE, and most other classes of nucleic acid reactivity experiments. The algorithms in ShapeFinder (1) convert measured fluorescence intensity to quantitative cDNA fragment amounts, (2)

[1] This work was published in *RNA* (2008) 14(10):1979-90. Reproduced with permission from Cold Spring Harbor Laboratory Press.

correct for signal decay over read lengths extending to 600 nts or more, (3) align reactivity data to the known RNA sequence, and (4) quantify per nucleotide reactivities using whole-channel Gaussian integration. The algorithms and user interface tools implemented in ShapeFinder create new opportunities for tackling ambitious problems involving high-throughput analysis of structure-function relationships in large RNAs.

2.2 INTRODUCTION

An absolute prerequisite for understanding the function of any RNA is an accurate picture of its higher order structure. Analysis of in-solution nucleic acid structural information often requires that RNA or DNA fragment lengths be analyzed at single nucleotide resolution. Important examples in this class include "footprinting", chemical modification and modification-interference experiments [8, 21-25] These experiments can be performed in a wide variety of ways designed to analyze local nucleotide conformational differences, solvent accessibility, and the effects of functional group modifications on RNA and DNA folding and interactions with protein and small molecule ligands. For over three decades, these classes of experiments have been evaluated by resolving nucleic acid fragments on polyacrylamide slab gels [26]. Gel electrophoresis has significant advantages including good nucleotide resolution of nucleic acid fragments and low material costs. However, gel electrophoresis is time consuming, single nucleotide-resolution separation is typically limited to 80-100 nts per gel, and band overlap and compression artifacts occur for many fragments.

In contrast to the limited read lengths obtained by gel electrophoresis, commercially available capillary electrophoresis instruments of the type commonly used for DNA sequencing routinely yield read lengths of 300 to 1000 positions at single nucleotide resolution. However, the absence of an appropriate set of software algorithms that address the unique quantitative properties of raw electropherograms generated by structure-probing experiments has prevented the use of capillary electrophoresis for high-throughput, single-nucleotide resolution, analysis of nucleic acid folding, dynamics, and ligand binding.

To address this problem, we have created a new software suite called ShapeFinder that automates the steps required to extract quantitative, single nucleotide resolution reactivity information for 300-650 nts in a single capillary electrophoresis run. We focus here on the analysis of SHAPE (selective 2'-hydroxyl acylation analyzed by primer extension) experiments [7, 8, 27]. However, the algorithms created in this work can also be used to analyze raw capillary electrophoresis data from other classes of nucleic acid reactivity experiments, including those that use other chemical modification agents or hydroxyl radicals to map structure and solvent accessibility (unpublished data).

2.2.1 RNA Structure and hSHAPE Chemistry.

SHAPE chemistry holds considerable promise for rapidly determining the structure of any RNA, under a variety of functionally important states, with single nucleotide resolution [28-37]. SHAPE chemistry involves measuring local backbone flexibility at nearly every position in an RNA by forming sparse 2'-*O*-adducts with using a hydroxyl-selective electrophile. This modification reaction can be made

exquisitely sensitive to local nucleotide flexibility [7] (Figure 2.1A). Nucleotides that are constrained by base pairing or tertiary interactions are unreactive, while conformationally flexible (and likely single-stranded) nucleotides preferentially form 2'-*O*-adducts (Figure 2.1A,B). Sites of modification are located by annealing a 5'-end labeled primer to the RNA and then extending the primer to the nearest site of modification using reverse transcriptase in an optimized primer extension reaction [7, 8]. The product of this experiment is a series of extended, 5'-end labeled cDNA fragments whose length and amount correspond to the position and degree of modification -- and hence local nucleotide flexibility -- at every nucleotide in an RNA (Figure 2.1C). In order to assess RNA degradation and position-dependent processivity of the primer extension reaction, a control omitting the reagent is performed in parallel. Third, in addition to the (+) and (–) reagent reactions, one or two dideoxy sequencing reactions are used to map reactivity to the RNA sequence (Figure 2.1D).

In high-throughput SHAPE (hSHAPE), each of the three components of a SHAPE experiment is implemented using the same primer sequence but labeled with a color-coded fluorophore. The cDNAs from the reactions are combined and run in a single capillary on a capillary electrophoresis sequencing instrument using procedures analogous to high-throughput DNA sequencing [6, 11, 38]. Raw capillary electrophoresis profiles typically contain 300-500 nts of SHAPE reactivity information and thus provide a comprehensive, nucleotide-resolution view of RNA secondary and tertiary structure in a single experiment. However, the raw electropherograms

are complex and require substantial processing before they can be used to infer RNA structural information.

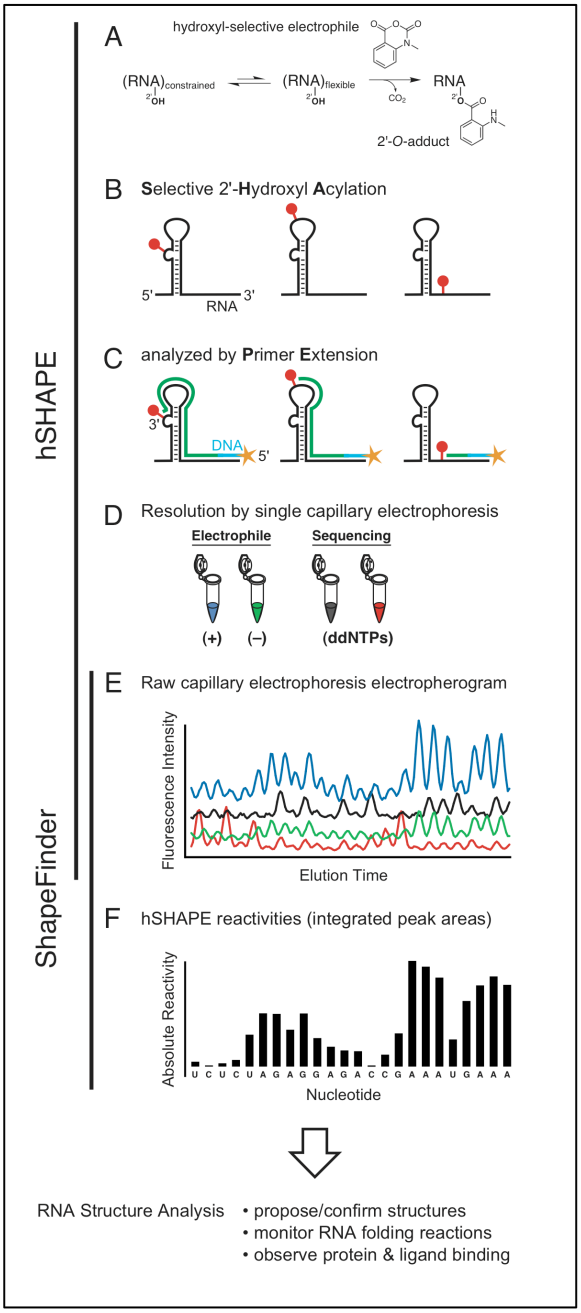

Figure 2.1. Overview of high-throughput Selective 2'-Hydroxyl Acylation analyzed by Primer Extension (hSHAPE) and data processing using ShapeFinder.

2.2.2 Algorithmic Challenges for Nucleic Acid Structure Analysis Resolved by Capillary Electrophoresis.

The output of an hSHAPE experiment resolved by capillary electrophoresis is an electropherogram, or trace. A typical trace contains 3 to 4 individual channels of fluorescence intensity versus elution time data; where each channel roughly corresponds to one of the SHAPE reactions (Figure 2.1E). The results of hSHAPE and DNA sequencing experiments resemble each other in that both experiments generate a series of measured fluorescence intensities versus elution time and must be processed extensively in order to yield useful nucleotide resolution information. However, extracting reactivity versus nucleotide position information for an hSHAPE or any other nucleic acid reactivity experiment requires the use of unique algorithms and data processing strategies.

The first and most important difference is that peak magnitude in DNA sequencing contains little meaning other than to indicate which nucleotide is present at a position. In contrast, both peak intensity and position are meaningful for all peaks in the (+) and (–) reagent channels in an hSHAPE experiment. Peak intensity spans a dynamic range of 50-fold and reports the structure-sensitive yield of the 2'-*O*-adduct, and thus local nucleotide flexibility (Figure 2.1A). The position reflects the length of the extended primer, and hence the nucleotide position in the RNA. Critically, the processing steps applied to hSHAPE data must not disturb relative intensity or distribution features of peaks in the electropherogram.

Second, 2-3 meaningful peaks in distinct channels [in the (+) and (–) reagent channels, and potentially in one sequencing channel] are observed per nucleotide in an hSHAPE electropherogram, versus one peak per nucleotide in a sequencing

experiment. Thus, hSHAPE peaks must be aligned with each other with greater precision because quantitative analysis of reactivity information requires greater alignment accuracy and it is not possible to base alignment on the expectation that there is only one intense peak per trace position.

Third, peak position and area must be determined for every position in the (+) and (–) reagent channels to quantify nucleotide reactivity; whereas, sequencing only requires locating the most intense peak per position. Importantly, the absence of a peak in the (+) reagent channel in an hSHAPE experiment represents significant information and indicates that a nucleotide is constrained by base pairing or tertiary interactions. Thus, accurate identification and quantitative analysis of noisy, barely detectable, peaks is an absolute requirement for successful hSHAPE analysis. Finally, fully automated analysis of hSHAPE data requires that sparse sequencing data be aligned to a known input sequence. This is the opposite of DNA sequencing, where the goal is to determine a precise sequence of nucleotides.

2.3 RESULTS

2.3.1 ShapeFinder

The initial processing steps required to convert raw capillary electrophoresis profiles into useful reactivity information are similar to those involved in analysis of DNA sequencing traces. We therefore extended the BaseFinder platform[9], a framework originally designed for DNA trace processing, analysis and base-calling, for analysis of nucleic reactivity information as resolved capillary electrophoresis. ShapeFinder is a modular, extensible software package in which each signal-processing algorithm is implemented as a tool. The results of each analysis step are

immediately displayed to the user in a straightforward graphical user interface (Figure 2.2).

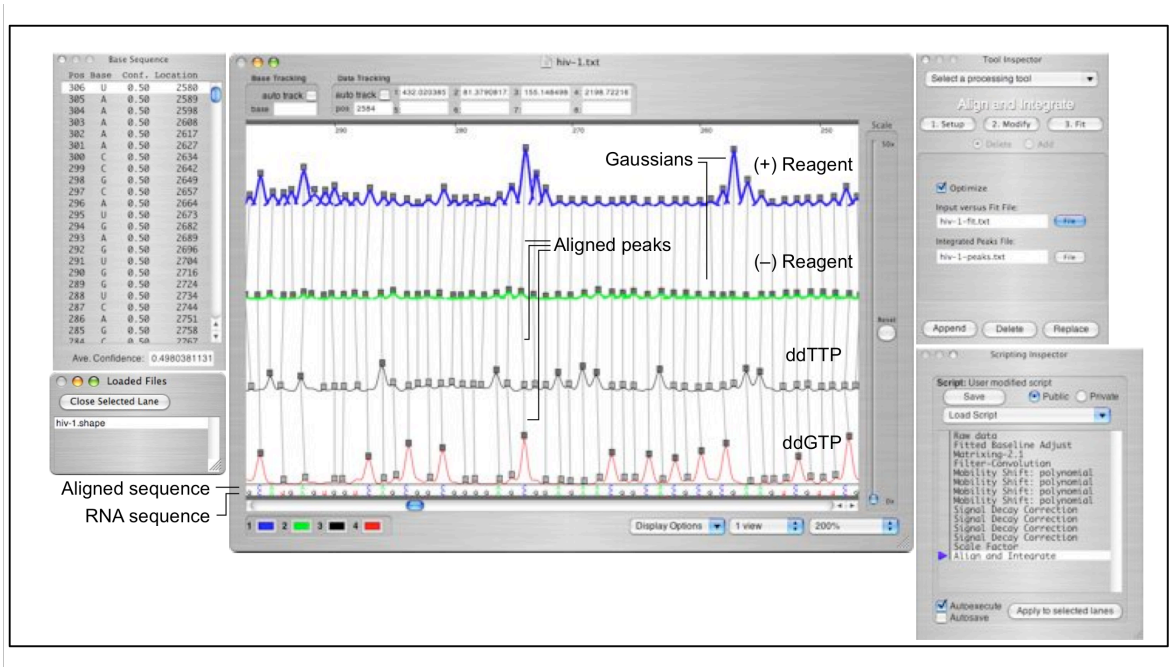

Figure 2.2. ShapeFinder at the Align and Integrate stage. The Data View Window (center) provides graphical feedback on each data processing step. The Tool Inspector window (upper right) displays the user-definable parameters for the tool selected in the Scripting Inspector. The Scripting Inspector (lower right) displays the tools thus far applied to the data.

ShapeFinder reads and displays files from most common sequencing platforms, including generic tab-delimited .txt files, the Beckman .esd and .dat files, and the ABI .fsa, .abi, and .ab1 formats. ShapeFinder also implements a new file format (.shape) that stores the raw and processed hSHAPE data along with the tool parameters that have been applied to the data set. The .shape file allows for review and re-execution of trace processing steps and facilitates testing the effects of different parameter choices.

The net output of ShapeFinder -- a table of quantitative reactivity information as a function of position in the nucleotide sequence (Figure 2.1F) -- can then be

used in numerous ways in the analysis of nucleic acid reactivity experiments. In the case of an hSHAPE experiment, reactivity information has thus far been used to develop models for an RNA secondary structure, to monitor RNA folding reactions, and to evaluate the effects of protein binding and macromolecular complex formation [6, 11, 38].

2.3.2 ShapeFinder Tools

ShapeFinder implements the algorithms required to convert raw capillary electrophoresis electropherograms into useful reactivity information through the execution of a specific sequence of tools, called a script. Each tool in a script accomplishes a specific data processing step by applying user-definable parameters to the electropherogram. The current script is displayed in the Scripting Inspector window in the ShapeFinder user interface (Figure 2.2, lower right). Tools are added and run using the Tool Inspector window, which also displays the parameter values associated with each tool (Figure 2.2, upper right). A processing tool is added using the "Append" button; tools already in a script may be changed and rerun by selecting "Replace." An individual step and its associated parameters may be reviewed by selecting the tool entry in the Scripting Inspector window.

Complete analysis of an hSHAPE raw capillary electrophoresis profile involves three major processing steps. First, the raw electropherogram is subjected to pre-processing to account for fluorescent background, correct for spectral overlap between the fluorescent channels, correct for the mobility shift imparted by tagging the primers used in the primer extension steps with different dyes, and adjust for signal decay at long read lengths. Second, channels are aligned so that all peaks in

the (+) and (–) reagent channels are identified and linked to the input RNA sequence, including those "peaks" corresponding to zero reactivity (peak alignment). Finally, quantitative nucleotide reactivities are obtained by performing a whole-channel Gaussian integration for all peaks in the (+) and (–) reagent channels (peak integration). Subtracting the integrated values for the (–) reagent from the (+) reagent profiles yields the absolute nucleotide-resolution reactivity for every RNA position over read lengths typically spanning 300-600 nts. An experienced individual can perform the data processing steps in approximately 1-2 hours.

We will illustrate these processing steps using an experiment performed on a transcript corresponding to the first 976 nts for the NL4-3 strain of the HIV-1 genome. Although any combination of detectable fluorophores or color-coding may be used to identify the individual reactions of an hSHAPE experiment, we will use a scheme in which blue and green channels represent the (+) reagent and (–) reagent experiments, and black and red represent RNA sequencing ladders (reflecting chain termination by ddGTP or ddTTP, respectively) (Figure 2.3). Data are collected from the capillary electrophoresis instrument such that the small fragments representing the 3'-end of the RNA read elute first.

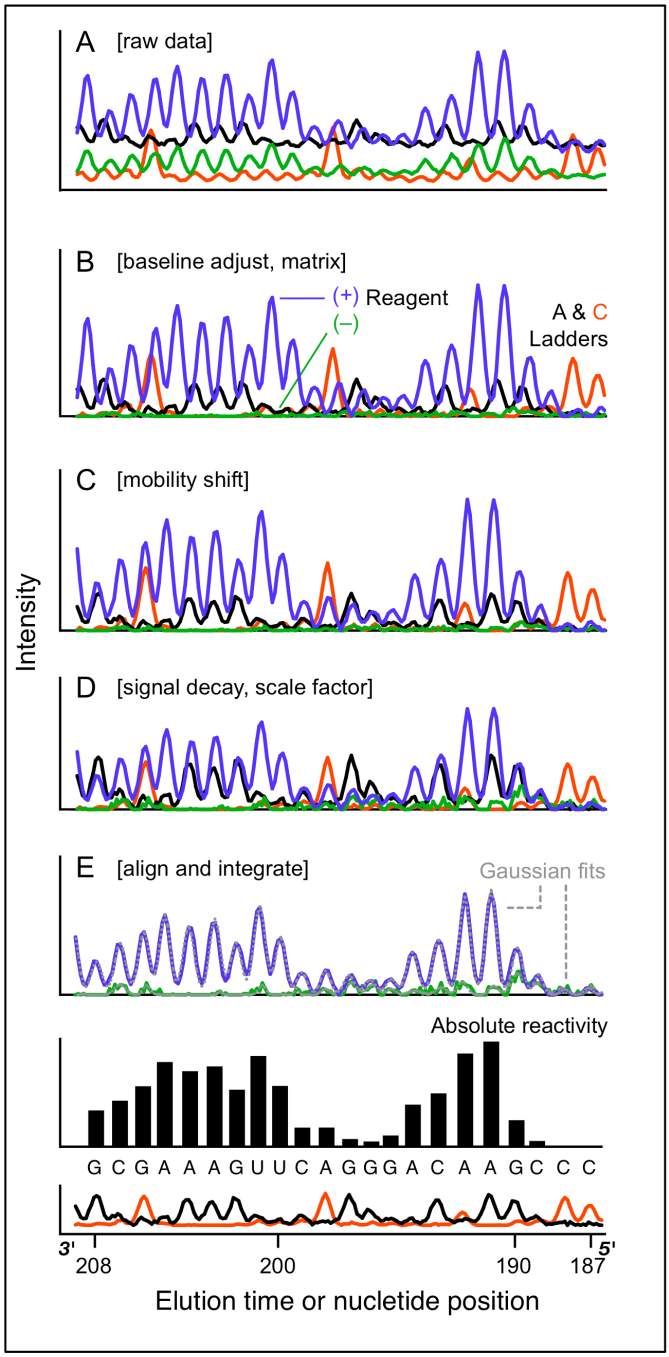

Figure 2.3. Electropherogram analysis as implemented using ShapeFinder tools. (A) Unprocessed capillary electrophoresis electropherogram. (B) Net result after application of the Fitted Baseline Adjust and Matrixing tools. (C) After the Mobility Shift: Polynomial tool (from four serial executions of the tool). (D) After Signal Decay Correction and Rescaling. (E) Whole-channel Gaussian integration of the (+) and (–) reagent channels obtained using the Align and Integrate tool. Solid bars show absolute SHAPE reactivities after subtracting background. For clarity, sequencing channels are offset from the (+) and (–) reagent channels.

2.3.3 Data Preprocessing

Fitted Baseline Adjust, Matrixing and Smoothing. Channels in raw capillary electropherograms are convoluted by detector background, overlapping emission spectra, detector noise, and horizontal offset between channels (Figure 2.1E, Figure 2.3A, and Figure 2.6A). Since these traits are common to all electropherogram data, initial processing of the raw electropherograms involves steps analogous to those used for DNA sequencing experiments.

Fluorescent background noise causes the baseline in each channel to drift, which imparts an idiosyncratic vertical offset to each channel. The Fitted Baseline Adjust tool adjusts each channel to a common baseline by zeroing each channel over a window of detector readings, typically ten times the average peak width.

The fluorescent dye used to distinguish the channels in a capillary electrophoresis electropherogram excite at similar wavelengths and have overlapping emission spectra. Thus, some dye signals are detected in several fluorescent channels by the instrument detector. For example, in the sample data set, the (+) reagent peaks are detected in both the blue and green channels (Figure 2.3A). The Matrixing tool determines the unique quantitative contribution of each fluorophore to signal intensity in each channel (Figure 2.3B). Parameters for the Matrixing tool must be calibrated once for each set of dyes (described in the Methods section). Some commercial instruments implement these steps using instrument-specific software and these alternative algorithms can be used in place of those in ShapeFinder, provided they correct completely for spectral overlap and do not leave significant residuals in other channels.

Trace data from a DNA sequencer contains fluctuations due to detector noise so that each major peak may have minor peaks and valleys of its own, which complicates downstream peak finding. Smoothing can increase read length and peak detection by ~10% for datsets with low signal to noise ratios. The result of the Baseline Correction, Matrixing and Smoothing tools on the HIV-1 NL4-3 transcript are shown in Figure 2.3B.

Mobility Shift. In an hSHAPE experiment, each reaction is analyzed using a DNA primer labeled with a different fluorophore (Figure 2.1D). The dyes alter the electrophoretic migration rate of the cDNA products so that cDNAs of the same length have slightly different elution times (Figure 2.3B). For hSHAPE data, correction for mobility shifts must be performed more accurately than is generally required for DNA sequencing to facilitate accurate location and linking of corresponding peaks between channels. ShapeFinder implements several mobility shift tools that can be combined in serial to account for horizontal offset without significantly altering peak shapes. Two to four serial applications of the Mobility Shift: Cubic tool typically places all channels on a consistent x-axis (Figure 2.3C). Parameters for an initial mobility shift must be set once for each set of primers. These parameters can also be fine-tuned on a trace-by-trace basis.

Signal Decay Correction. Inspection of all of the channels in an hSHAPE experiment indicates that peak intensities decay with increasing read length (best visualized in Figure 2.6A). There are three sources of this decline, depending on the reaction channel. (1) Reverse transcriptase is not perfectly processive and fails to elongate at every position with an unmodified 2'-hydroxyl, such that the probability of

adding an additional nucleotide to a cDNA is slightly less than one. (2) The (+) reagent reaction is designed that, on average, only one in every 300 nts is modified. However, because modification is random over long RNA lengths, some RNAs react two or more times. For RNAs containing multiple adducts, only the first site of modification is detected, thus favoring short cDNAs. (3) In the sequencing reactions, the population of extending primers decreases by a small factor each time a dideoxynucleotide is incorporated. Thus, the signal decays exponentially to zero in all channels at read-lengths of 400-650 nts. This decay is also observed in DNA sequencing experiments and is corrected by normalizing peak intensities across all channels to consistent heights [9].

In the case of hSHAPE experiments, fluorescence intensity is meaningful, so the decay correction must be performed using a statistical model of signal decay. We find that signal decay is well modeled as:

$$D(x) = Aq^x + C \quad (1)$$

where D is the signal intensity as a function of primer elongation, A is the amplitude of the decay, C represents the measured intensity at the end of the channel, and q is the probability of extension at position x [30].

The user sets (i) the range of data points, (ii) the channel which to apply the tool, and (iii) a scaling factor to maintain the scale of the corrected channel relative to the other channels. The algorithm calculates new values with roughly even reactivities over a 300-650 nt read length. A properly corrected channel is readily verified by visually inspecting the data: intense peaks in the beginning, middle and end of the channel should be of uniform height.

Scaling. Experimental variations in performing primer extension, inherent differences in dye intensity (compare black and red channels, Figure 2.6A) and second-order effects of the ShapeFinder Smoothing and Signal Decay Correction tools can cause the channels to be on different scales. The channels are adjusted manually such that the smallest 5-10% of the peaks throughout the (+) channel match corresponding (–) channel intensity (Figure 2.3D, compare blue and green channels). This correction assumes that there are always a few completely unreactive nucleotides in an hSHAPE read whose peak intensities should exactly match the corresponding (–) peak intensities. For ease in data viewing and further analysis, sequencing peaks are set to match moderately intense peaks in the (+) channel (Figure 2.3D, compare red and black to blue channels). After preprocessing, all channels have a baseline set to zero, peaks in different channels corresponding to the same nucleotide have the same elution time, and well-defined peak intensities correspond quantitatively to cDNA amounts (Figure 2.3D).

2.3.4 Whole-Channel Peak Alignment and Integration

The heart of the new ShapeFinder program is the Align and Integrate tool, which calculates hSHAPE reactivities for every analyzable nucleotide in an electropherogram (Figure 2.3E). There are four phases to the algorithm: (1) peak finding and linking, (2) alignment to the RNA sequence, (3) user editing of the alignment by adding or deleting peaks, and (4) quantification of 2'-*O*-adduct formation by peak-by-peak Gaussian curve fitting (Figure 2.4). The ShapeFinder Align and Integrate tool implements these steps in the Setup, Modify and Fit panels (Figure 2.2, upper right). Phases 1 and 2 are performed using the Setup panel; the

Modify panel allows the user to add and remove peaks in phase 3; and the Fit panel is used to manage phase 4. The tool is iterative and alignments are recalculated after each round of user input. Subtracting the (–) from the (+) reagent peaks is performed automatically and yields absolute hSHAPE reactivities for every nucleotide in the capillary electrophoresis read (bars, Figure 2.3E).

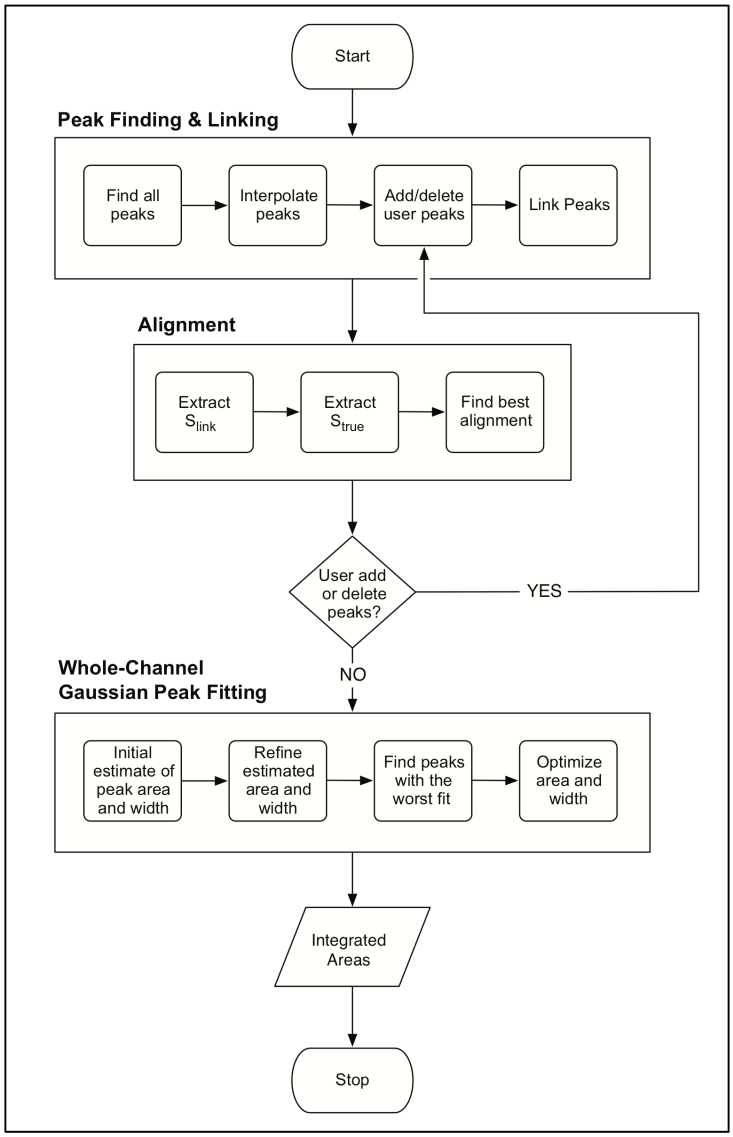

Figure 2.4. Flow chart of the Align and Integrate algorithm, which involves three phases: (1) peak finding and linking of (+) reagent, (–) reagent, and sequencing peaks, (2) peak alignment to the RNA sequence, and (3) calculation of per nucleotide SHAPE reactivities by Gaussian integration.

Primer extension stops at the base preceding the nucleotide containing a 2'-*O*-adduct; thus, the (+) and (–) reagent peaks are one nucleotide longer than the cDNA fragments generated by dideoxy sequencing [7]. Therefore, the sequencing alignment is shifted by one nucleotide relative to the (+) and (–) reagent channels. To avoid confusion, the ShapeFinder display shows the sequencing peaks without an offset. Text output files show the offset.

Setup. In the Setup phase, the user assigns each channel to one of the four SHAPE reactions [(+) and (–) reagent, and sequencing ladders] and specifies the region of the trace to be analyzed using either numerical trace positions or by selecting a region of the trace in the main window. A Refine option enables automatic interpolation of peaks in the (+) reagent or (–) reagent channels based on the expected spacing in a given region of the trace. The Setup phase also reads a text file containing the RNA sequence in the 5' to 3' direction that is used to align the trace data to the RNA nucleotide position.

A preliminary alignment is initiated after these parameters have been set. The data view window displays the four channels and demarcates identified peaks with squares (Figure 2.2). Vertical lines show peaks in the four channels that have been linked with each other and with the input sequence. Light-shaded squares in either the (+) or (–) reagent channels indicate unlinked peaks. For the sequencing channels, light-shaded squares report peaks that were identified but were not accepted as part of the sequencing ladder.

Modify. In portions of a run with strong signal, low-noise, and good alignment, the results of the first alignment step are typically satisfactory. However,

in some regions, especially near the ends of a read, meaningful peaks may be missed, not aligned, or assigned incorrectly. However, these regions often contain high quality and quantitative SHAPE structural information that can be gleaned with operator supervision. To this end, ShapeFinder allows manual editing and extension of the automatically generated alignment using the Modify panel (illustrated schematically in Figure 2.4).

The ShapeFinder-determined sequence alignment is displayed at the bottom of the data window (Figure 2.2). The top sequence is determined from the sequencing channels, while the bottom sequence shows the loaded sequence. For completely aligned data, letters coincide vertically between the two datasets. When the data are partially misaligned relative to the input sequence, there will be a horizontal offset between the two sequences. The addition or deletion of a peak in the (+) reagent channel is usually required to correct the alignment. Finding the location of a missed or incorrectly added peak is accomplished in a straightforward way by locating the position where horizontal offset begins.

Peaks to be deleted or added are selected by clicking on the square at the top of each peak, or clicking at the desired position for a new peak in any channel, respectively. The tool window displays a spreadsheet-style list of peak positions that have been added or deleted (left panel, Figure 2.2). Data that is ready for Gaussian fitting (center, Figure 2.2) is correctly aligned to the sequence and has all (+) and (–) peaks linked to each other and to the sequence, as indicated by filled boxes. Depending on the quality of the data, zero to ~20 peaks may need to be deleted or added at either end of a trace to allow a complete alignment. Particularly difficult or

unalignable regions may be removed in the Setup panel by adjusting the Trace Range. For the HIV-1 example dataset, three unaligned (+) reagent peaks were deleted at the beginning of the trace. This correction then allowed complete alignment of >400 continuous nucleotides in the RNA.

Fit. Once all peaks to be analyzed have been identified and linked with the input sequence, the Fit phase of the Align and Integrate tool performs whole-channel Gaussian peak integration for the (+) and (–) reagent channels (Figure 2.3E). Each peak is fit to

$$y_i(x) = \frac{A_i}{\sqrt{2\pi\sigma_i}} e^{-\frac{1}{2}\left(\frac{x-\mu_i}{\sigma_i}\right)^2} \quad (2)$$

where A_i is the peak area and μ_i and σ_i are the center and width of peak *i*, respectively. The tool has both fast and optimize modes. The Optimize option provides a more accurate peak fitting, but is more computationally demanding (Figure 2.5).

Once fitting is complete, ShapeFinder displays the calculated peaks superimposed upon the (+) and (–) reagent channels in the data view window (Figure 2.2, bold lines). The fitted Gaussian curves for all peaks, the calculated peak areas, the net absolute reactivity at every position, the alignment to the input sequence, and identified peak positions are output in tab delimited ASCII text files.

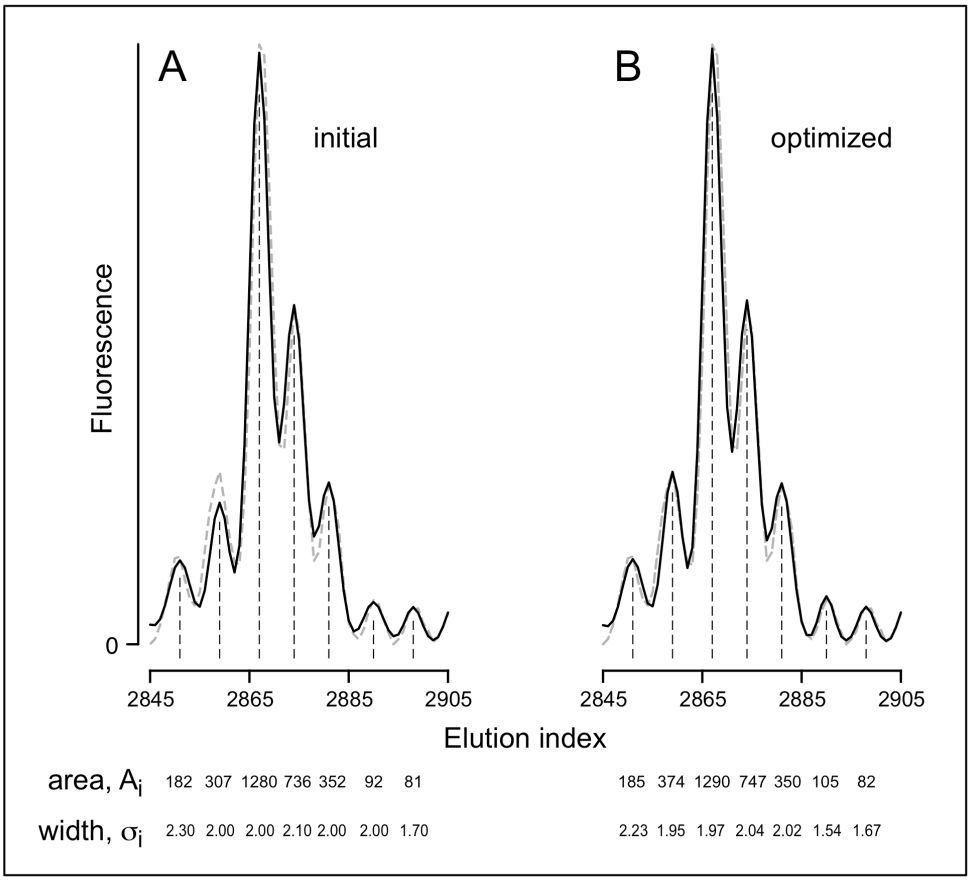

Figure 2.5. Whole-channel peak integration. (A) Preliminary local fit to initialize values for A_i and σ_i. (B) Final globally optimized fit.

2.3.5 Example of a Complete hSHAPE Experiment, Quantified by ShapeFinder

A complete hSHAPE electropherogram contains structural information for several hundred RNA nucleotides (Figure 2.6A). As outlined above, the raw data for the HIV-1 example RNA includes all of the typical characteristics of raw electropherogram sequencing data, including baseline offset, fluorescent overlap, channels on different intensity scales (black and red channels, Figure 2.6A), mobility offset for cDNAs of the same length, and signal decay such that peaks at the left of the channel are 4 times more intense as those at the right (blue and red channels, Figure 2.6A). After applying the preprocessing tools, all channels have a baseline

set to zero, peak intensities correspond quantitatively to cDNA amounts, and overall peak heights are distributed evenly throughout each channel (Figure 2.6B).

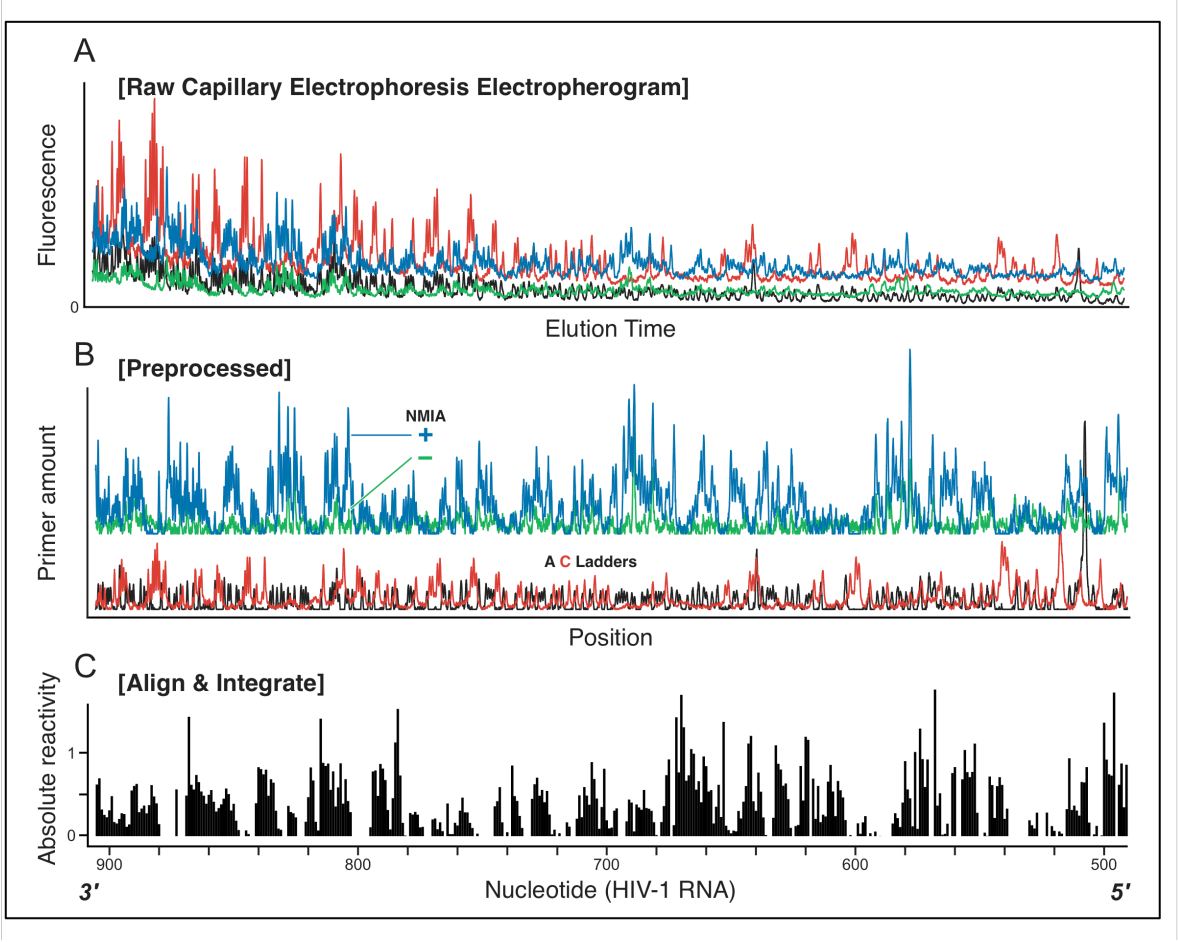

Figure 2.6. Overview of a complete hSHAPE data set, processed using ShapeFinder and representing a total read length of 415 nts from an HIV-1 transcript RNA. (A) Raw electropherogram from a DNA sequencer. The data consists of four channels of fluorescence intensity information as a function of elution time. (B) Preprocessed SHAPE data. Each channel now represents dye amount, not fluorescence, as a function of elution time for each of the four channels. For clarity, channels corresponding to the A and C sequencing ladders are offset from the (+) and (–) reagent channels. (C) Sequence alignment and whole-channel Gaussian peak integration using the Align and Integrate tool to calculate absolute SHAPE reactivities.

ShapeFinder then implements a whole-trace Align and Integrate algorithm that (i) aligns the sequencing ladders and the (+) and (–) reagent channels with the input sequence and (ii) calculates the areas under all analyzable peaks in the (+)

and (–) channels by whole-trace Gaussian integration. Subtracting the (–) peak areas from the (+) reagent peaks yields the absolute hSHAPE reactivity for every nucleotide in the capillary electrophoresis electropherogram (bars, Figure 2.6C). In this typical experiment, single nucleotide resolution SHAPE reactivities were obtained for positions 491–905 in the HIV-1 transcript, for a total read length of 415 nts.

2.3.6 Analysis of Accuracy and the Reproducibility of hSHAPE and ShapeFinder

The most important criteria by which to judge ShapeFinder is whether its algorithms yield reproducible and accurate RNA structure information. We first analyzed SHAPE reactivities for the well-studied $tRNA^{Asp}$ molecule. After performing a SHAPE experiment on $tRNA^{Asp}$, cDNA fragments were resolved either (i) using radiolabeled DNA primers and detection by denaturing gel electrophoresis or (ii) by capillary electrophoresis and ShapeFinder. The tRNA molecule was imbedded in a previously described structure cassette to facilitate analysis by primer extension [7]. The net length of this RNA is 132 nts, which is close to the limit in RNA size that will yield a single-nucleotide resolution banding pattern in routine sequencing gels. cDNAs resolved by gel electrophoresis were quantified using SAFA, which has been independently validated to calculate accurate band intensities [39]. We compare these experimental results to a SHAPE experiment performed under the same conditions on the same RNA, but analyzed using fluorescent primers, capillary electrophoresis, and ShapeFinder. For both datasets, SHAPE reactivity data are normalized to a scale that spans 0 to ~1.5 and in which 1.0 is defined as the average reactivity of highly reactive positions.

Inspection of the two datasets indicates that quantitative analysis of cDNA fragments obtained from a SHAPE analysis of the $tRNA^{Asp}$ transcript yielded nearly identical reactivities at almost all positions, regardless of the separation and analysis platforms (compare solid and open columns, Figure 2.7A). The linear relationship correlation coefficient, R, between the two datasets is 0.91, indicating 83% (R^2) of the variability of the hSHAPE data can be predicted by the variability of the gel data. This correlation is significant at the $p < 0.0001$ level. Comparison of the differences in reactivities between the two datasets yielded a Student t-test p-value of 0.84, indicating the group reactivities are statistically equivalent.

The only significant differences in measured nucleotide reactivity occurred at positions 29-32. The differences reflect the difficulty in calculating intensities in the context of band compression that occurs when cDNA fragments for this RNA are resolved by gel electrophoresis [27, 40] (labeled, Figure 2.7A); these positions were therefore not included in the correlation analysis. In contrast, positions 29-32 were readily interpretable in the capillary electrophoresis trace. Thus, ShapeFinder yields quantitative values for per nucleotide reactivities that are as accurate as the conventional approach using gel electrophoresis. The only difference is that capillary electrophoresis is less sensitive to band compression artifacts.

Second, we analyzed the reproducibility of SHAPE reactivities for five data sets, three corresponding to a primer binding at position 342 and two for a primer binding at position 535 (Figure 2.7B). These primers bind 193 nts apart and yield overlapping reads of ~200 nts. The region of overlap corresponds to the 3' portion of one primer read, and the 5'-most end of the second primer read (see dashed

arrows, Figure 2.7B). The overlapping regions therefore also correspond to sets of peaks that have been differentially adjusted by the Signal Decay Correction algorithm.

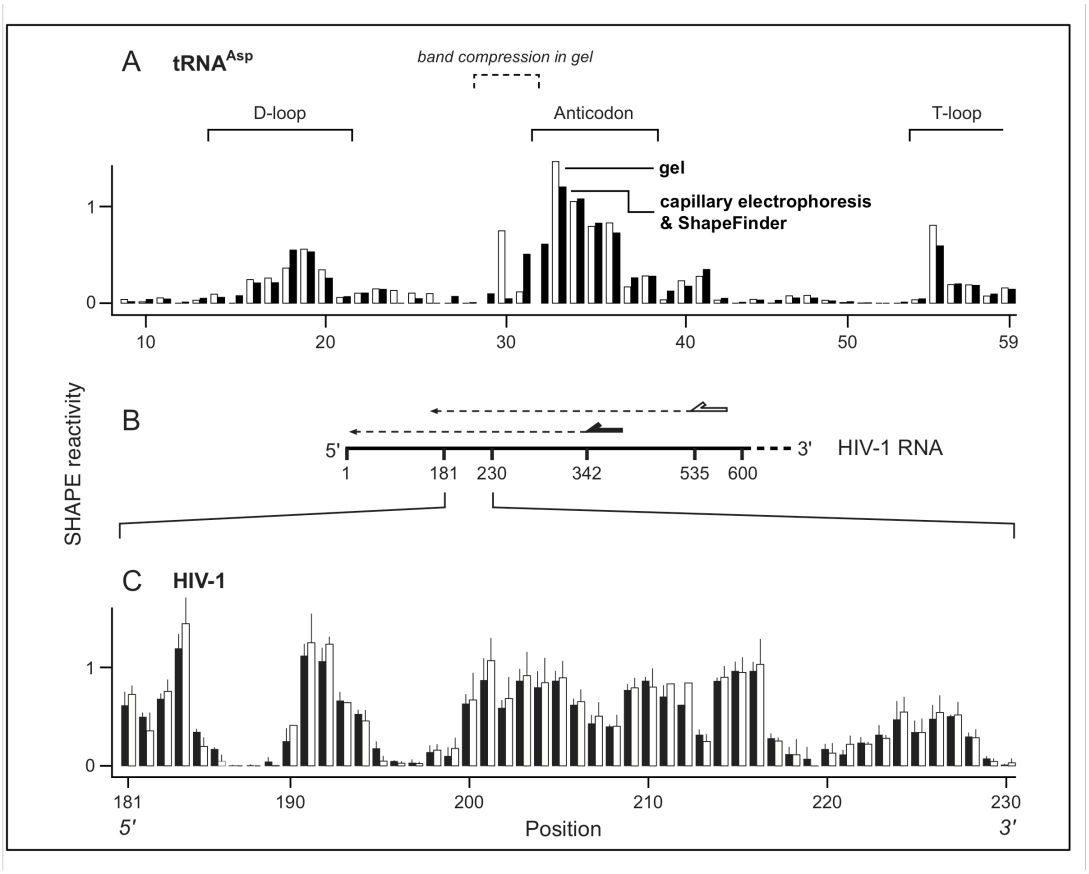

Figure 2.7. Accuracy and reproducibility of hSHAPE and ShapeFinder. (A) Comparison of nucleotide reactivity as quantified by ShapeFinder (closed bars) and denaturing gel electrophoresis (open bars). Loops in $tRNA^{Asp}$ are indicated explicitly. The $tRNA^{Asp}$ sequence was flanked by 5' and 3' structure cassette sequences [7]. Due to strong band compression [27, 40], some positions cannot be visualized by gel electrophoresis. Bands visualized by gel electrophoresis were quantified using SAFA [39]. (B) Overlapping reads for HIV-1 genome transcripts. Primers, shown as solid and open arrows, anneal to the RNA 193 nts apart and reads therefore overlap by ~200 nucleotides (dashed lines). (C) Mean hSHAPE reactivities and standard deviations calculated from overlapping and replicate reads. Primers annealed at positions 342-363 (solid columns) and 535-555 (open columns). Data shown report three experiments from the 342-363 primer and two experiments from the 535-555 primer for five experiments total, whiskers report standard deviations. Due to high background, no data was available at nt 219.

We performed several statistical tests to evaluate how similar calculated peak intensities are across data sets. Correlation coefficients calculated between the 10 possible pairs of the 5 datasets indicated a very strong correlation between the datasets, with R^2 values ranging from 0.86 to 0.97 (p-values < 0.0001). A one-way ANOVA (analysis of variation) performed between the 5 datasets showed the SHAPE reactivities to be statistically equivalent (p = 0.77). Furthermore, Levene's Test indicated constant variance between the five datasets (p-value = 0.26). Finally, we calculated the standard deviation for each measurement in the 181-230 window. A plot of the per position standard deviation as a function of mean SHAPE reactivity is linear (R = 0.73; p < 0.0001). Linear regression indicates that the average measurement error at any one nucleotide is 0.04 + 0.11 × (per position measurement) in SHAPE units. Thus, for representative low and high SHAPE reactivities of 0.1 and 0.7, measurement errors are expected to be ±0.05 and ±0.12 SHAPE units, respectively.

In sum, these statistical tests indicate that SHAPE reactivities as quantified using ShapeFinder (i) are calculated accurately over hundreds of nucleotides, (ii) are accurately corrected for signal decay as modeled by Eqn. 1, and (iii) exhibit small absolute measurement errors. Combining quantitative reactivities from individual reads of 300-650 nts can therefore robustly monitor the structures of long RNAs, potentially spanning thousands of nucleotides.

2.4 Discussion

Experiments that probe nucleotide reactivity and solvent accessibility represent powerful approaches for analyzing conformational changes and protein

and ligand binding for RNAs of known structure, and for developing models for RNAs whose structures are not known. A critical limiting step in such analyses has been the use of gel electrophoresis technology to visualize the results of these experiments. In many cases, more effort is required to obtain, manipulate, and quantify information by gel electrophoresis than is spent actually performing the experiment or interpreting its result.

The algorithms implemented in ShapeFinder dramatically lower the barriers to monitoring the structure of large RNAs. Depending on the characteristics of an RNA, we routinely obtain read lengths of ~400 nts, with reads of up to 650 nts under optimal conditions. This means that single nucleotide resolution structure information can now be obtained for entire large catalytic and regulatory RNAs or domains of larger RNAs like ribosomal RNAs in a single experiment.

While ShapeFinder accelerates the ability to interrogate RNA structure in solution at single nucleotide resolution, we are continuing to develop new methods and algorithms to further improve the speed, accuracy and automation of hSHAPE analysis. Important objectives include improved and automatic mobility shift alignment and algorithmic optimizations to reduce the computational overhead for peak curve fitting. It is our hope that ShapeFinder will make it possible to tackle new classes of problems related to the role of long-range RNA structure in biological function.

2.5 MATERIALS AND METHODS

2.5.1 SHAPE Data.

SHAPE experiments were performed on an HIV-1 transcript or $tRNA^{Asp}$ exactly as described [6, 7]. Most of the SHAPE reactivity data on HIV-1 sequences presented here was reported previously [6]. Briefly, DNA templates encoding the 5'-most 976 nts of the HIV-1 NL4-3 strain (Gen Bank AF324493) or $tRNA^{Asp}$ were generated by PCR. The RNA construct was produced by *in vitro* transcription and purified by gel electrophoresis. The HIV-1 RNA and $tRNA^{Asp}$ (1 pmol) were refolded in 50 mM HEPES (pH 8.0), 200 mM potassium acetate (pH 8), and 5 mM $MgCl_2$; or 100 mM HEPES (pH 8.0), 100 mM NaCl, 10 mM $MgCl_2$, respectively, at 37 ˚C for 30 min. (+) and (–) reagent SHAPE reactions were initiated by treating the RNA with N-methylisatoic anhydride (NMIA, 32.5 mM in DMSO) or DMSO, respectively. After the NMIA hydrolyzed completely (60 min) [7], the RNA was recovered by ethanol precipitation and mixed with fluorescently labeled DNA primers (Proligo or LI-COR) that annealed at either positions 342-363, 535-555 or 956-976. Primer extension was initiated by addition of Superscript III reverse transcriptase (Invitrogen). Sequencing markers were generated using unmodified RNA by performing primer extension in the presence of dideoxy NTPs. Dyes for the (+) and (–) reagent and sequencing lanes were Cy5, WellRed D3, WellRed D2 and LI-COR IR 800, respectively. cDNA products from the four reactions were mixed, purified, and separated on a Beckman CEQ2000XL capillary electrophoresis DNA sequencer. $tRNA^{Asp}$ experiments were performed both using a 5'-radiolabeled primer and resolving primer extension fragments on electrophoresis gels [7, 27] or using fluorescently labeled (Proligo or LI-COR) primers and capillary electrophoresis, in

the same manner as HIV-1, except that 130 mM NMIA was used. Fluorescence intensity over the 4 channels was monitored at a rate of 2 Hz and yielded an average of ~10 points per peak position. Raw electropherograms were output from the capillary electrophoresis instrument in the Beckman .txt format and read directly into ShapeFinder.

2.5.2 ShapeFinder Software

ShapeFinder is a derivative of the BaseFinder trace-processing platform and is written in Objective-C [9]. It is distributed as a Universal Binary, and runs on Macintosh PowerPC or Intel computers running Max OS X 10.4 or later. ShapeFinder is freely available for non-commercial use. A comprehensive Help File is also available in the software package for new users of hSHAPE technology and ShapeFinder. Both the ShapeFinder software and help package, as well as all HIV-1 data and example scripts used in this work are available at: http://bioinfo.unc.edu/downloads/.

Fitted Baseline Adjust. The Fitted Baseline Adjust tool calculates a common baseline for each channel while keeping the experimentally recorded data intact [9]. The local minima in a channel are found after dividing the channel into windows representing 5-20 times the average peak width. For the HIV-1 dataset, the window size was 200 because peak widths usually are ~10±5 data points.

Matrixing. The multiple fluorescent dyes excite at different wavelengths and have overlapping emission spectra, such that each channel contains contributions from more than one dye in multi-fluor runs. Spectral overlap was removed using a linear transformation matrix so that each channel represents dye amount as a

function of position. The transformation matrix is calibrated using four extension reactions run in separate capillary columns, which need be performed only once per dye set. The extension reactions must generate a series of intense, but not saturating, peaks for each fluorophore, and is most easily achieved by generating sequencing channels. The extension products are resolved in independent capillary runs, such that each electropherogram contains fluorescence from a single dye. The user selects an intense peak for each of the dyes and ShapeFinder automatically calculates a transformation matrix that can be used for all experiments using the same dye set.

Smoothing. Trace data from a DNA sequencer contains fluctuations due to detector noise. The peak-fitting and alignment algorithm implements an internal smoothing step to correct such noise and we find that this smoothing is sufficient for optimal processing in most cases. However, in cases where trace data are very noisy, or the user prefers the displayed data to be smoothed, a separate smoothing step can be applied using the Filter-Convolution [9] tool. Recommended parameters are a Gaussian width σ = 1 and window size of 10. Judicious noise reduction by smoothing is helpful; however, it is important that this step not be overdone or adjacent peaks can blend together.

***Mobility Shift*.** Mobility shift parameters are calculated using a sequencing experiment in which all four dye labeled primers are extended in the presence of the same dideoxy nucleotide and resolved in a single capillary. Tool parameters are initialized by dragging portions of channels so that they all align to a user-chosen reference channel. The algorithm calculates coefficients to fit a polynomial equation

to the data. These parameters need be determined only once for a given primer set, but individual electropherograms may require fine-tuning. Two to three iterations of the Mobility Shift: Cubic tool are usually sufficient for an hSHAPE electropherogram.

Signal Decay Correction. This tool corrects the decay in peak intensity due to the stochastic nature of 2'-hydroxyl modification and the imperfect processivity of reverse transcriptase. At each nucleotide position, there is a probability *p* that a reagent-modified nucleotide will stop reverse transcriptase. The probability that the reaction will continue, *q* [$q = (1 - p)$], yields the exponential form observed for peak drop-off that we model with Eqn. 1.

For each hSHAPE experiment, the algorithm determines the best-fit parameters from equation 1 in two steps. First, it identifies peak locations, calculates their height, and removes outliers. Peaks are identified by considering seven consecutive points, calculating the slope of the line connecting each sequential pair of neighboring points, and then averaging the six consecutive slopes. Peak maxima are identified as the points where the derivative transitions from positive to negative. Anomalously high outlier peaks are identified and excluded using a box plot model in which outliers fall outside 1.5 times the inter-quartile range of the data [41]. Second, the algorithm fits the remaining peak heights to Eqn. 2 to determine *A*, *C* and *q* using Levenberg-Marquardt non-linear least squares parameter estimation [42, 43]. The probability of extension, *q*, is typically ~0.999 for a broad range of datasets, whereas *A* and *C* vary significantly due to the arbitrary instrument units that describe fluorescence intensity. Correction of the HIV-1 dataset yielded coefficients *A=0.09*, *q=0.9994*, and *C*=0.002 for the (+) reagent channel.

Each channel in the hSHAPE electropherogram is corrected independently for signal decay.

After parameter estimation, the reagent and control channels are then corrected for signal decay using:

$$I_{new}(x) = N \times I_{old}(x)/D(x) \quad (3)$$

where I_{old} is the original measured intensity, $D(x)$ is from Eqn. 1, and N is a user-definable rescaling factor that maintains overall peak intensity relative to the other non-corrected channels.

Scale Factor. This simple tool is used to rescale individual channels in a trace. This may be necessary because fluorescent intensity values measured in distinct channels depend on the properties of the fluorophores and detector, resulting in an arbitrary relative scaling between channels. The tool takes the user-specified channel scaling factors and multiplies them against the intensity values for the specified channel, adjusting each channel to be on the same relative vertical scale. Data should be scaled so that peak heights range above 100 (arbitrary) units to improve the accuracy of subsequent peak fitting.

2.5.2.1 Align and Integrate

(1) Peak finding and linking. This first phase accepts user input that specifies (i) which extensions [(+), (–), or sequencing] were performed in each channel, (ii) the region of the preprocessed data to analyze, and (iii) the sequence. The sequence is read from an ASCII-formatted text file where white space characters, non-A, G, C, U, T or N characters, and FASTA headers are ignored. Once loaded, the sequence is reversed to correspond to the SHAPE experiment, in

which cDNAs elute in the 3' to 5' direction with respect to the RNA sequence. Prior to peak finding, data are smoothed over a three-point window (dashed lines Figure 2.8). The algorithm first identifies the peaks in each channel by identifying those that are the highest point centered within a range of ±3 neighboring points. The most frequent distance between peaks is then calculated, and additional peaks interpolated when the distance between two peaks is greater then the most frequent peak distance (initially identified and interpolated peaks are shown by the open and closed circles in Phase A of Figure 2.8). Since the linearly interpolated location may not lie on a local maximum, interpolated peaks are then shifted left or right to a maximal position [for example, position 2845 in the (+) channel in Phase B of Figure 2.8]. Peaks in the (+) and (–) reagent channels are aligned with each other if they are positioned near each other on the elution time axis, defined by a distance threshold *t* that is iteratively incremented from zero to a maximum value *k/2*, where *k* is the median distance between neighboring peaks in the channel. In sum, this algorithm matches the best-aligned peaks first and then incrementally allows for misalignment of peaks that do not initially match (illustrated by lines linking the circles in Figure 2.8). A preliminary alignment for 600 nts requires 2 seconds on a 1.5 GHz Power PC processor.

In some cases, there may remain unlinked peaks in the (+) and (–) reagent channels. The Setup phase implements a Refine option that creates a peak in the (-) reagent channel when a matching peak is not found for a (+) reagent peak (Phase C in Figure 2.8; for example, position 2627). Also, if a matching (+) reagent peak is not found for an identified (–) reagent peak, the (–) reagent peak is automatically

deleted. When the Modify option is used to add or delete peaks (see Figure 2.4) the algorithm adds these peaks and automatically creates the appropriate new peaks links (Phase D in Figure 2.8).

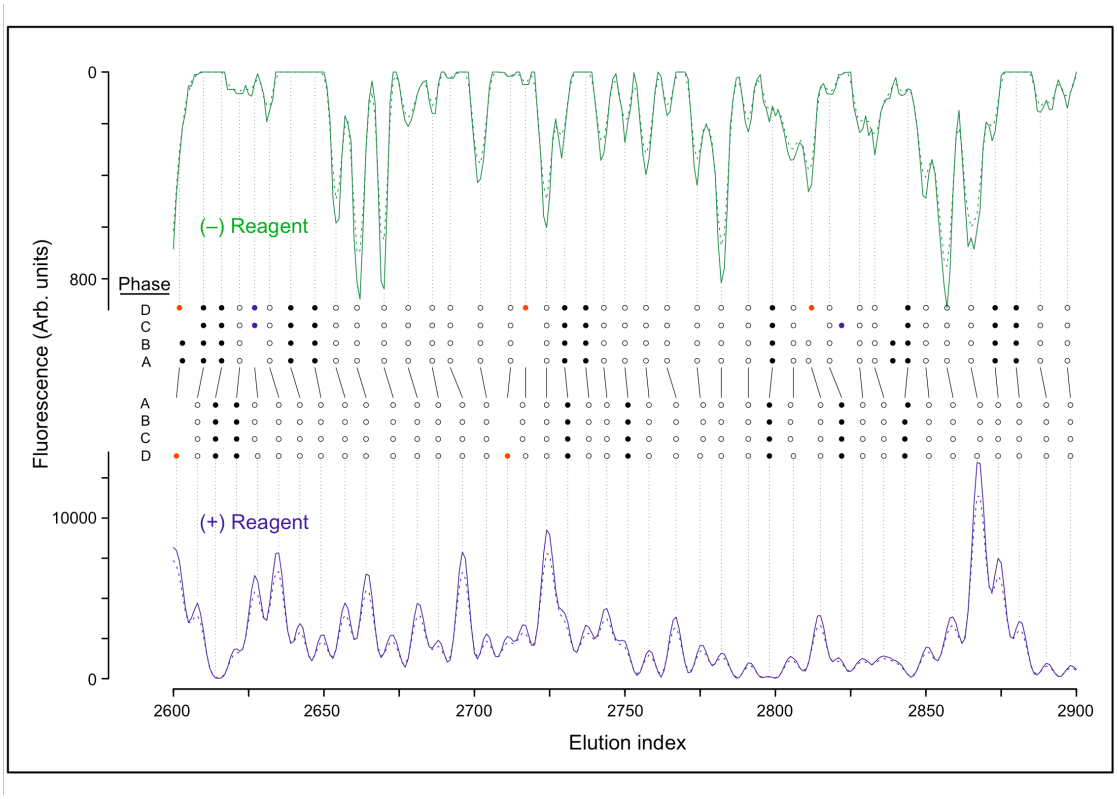

Figure 2.8. Peak finding. The electropherogram shows the (+) and (–) reagent channels (blue and green, respectively). The (–) reagent intensities have been inverted and are plotted on an expanded scale to facilitate visualization of peak synchronization. Preprocessed channels are shown as solid lines, channels smoothed over a 3-nt window are dashed. (A) Identification of peak positions by analysis of (i) signal amplitude and (ii) interpolation are illustrated by open and closed circles, respectively. (B) Refinement of interpolated peaks positions. (C) Automatic addition of missing peaks (blue circles) after comparison of the (+) and (–) reagent channels. (D) Incorporation of peaks added (red circles) or deleted by the user and subsequent refinement of peak positions. Positions of synchronized (+) and (–) reagent peaks that will be used during the integration phase are emphasized with solid lines.

Peaks in the sequencing channels are identified and aligned in a similar fashion (Figure 2.9, Phases A and B). Sequencing channels contain two classes of peaks: small background peaks due to imperfect processivity of the reverse

transcriptase and RNA degradation, as well as intense peaks indicating the presence of the sequenced nucleotide. These two classes are separated using a user-definable sensitivity level: peaks are part of the sequencing ladder only if their height is greater than the median channel intensity multiplied by the sensitivity level. By decreasing the sensitivity, more sequencing peaks are identified as part of a sequencing ladder; conversely, by increasing the sensitivity, fewer sequencing peaks are found. In the final step of this phase, sequencing peaks are linked to the (+) and (–) reagent peaks if the peak is within ±2 points on the x-axis of a (+) or (–) reagent peak (Figure 2.9, Phase C).

(2) Alignment to the RNA sequence. The next step in this algorithm is to align the trace data with the known RNA sequence (see Alignment steps in Figure 2.4). A sequence, S_{link}, is derived by correspondence of the sequence ladder from the ddNTP channels to the (+) reagent peaks, in which an N indicates the positions of non-sequenced nucleotides and the appropriate A, G, C, or U indicates the linked ddNTP peak. An example sequence for a reaction using ddUTP and ddGTP might be NCAANCNNNCNCAC. A sequence S_{true} is created from the known RNA sequence by including the positions corresponding with the sequenced nucleotides in the hSHAPE experiment, and replacing the non-sequenced nucleotides with N. S_{link} is then compared with S_{true} by sliding S_{link} along the length of S_{true}. The algorithm accepts the alignment that contains the most matching positions between S_{link} and S_{true} (illustrated in Phase C of Figure 2.9). In Figure 2.2, S_{true} is at the bottom of the data window (as the Input sequence) and S_{link} appears immediately above (Aligned sequence). Comparing the two sequences readily identifies a correct alignment.

Consistent horizontal offsets signify an incorrectly identified peak or unidentified peaks and are corrected by editing the alignment.

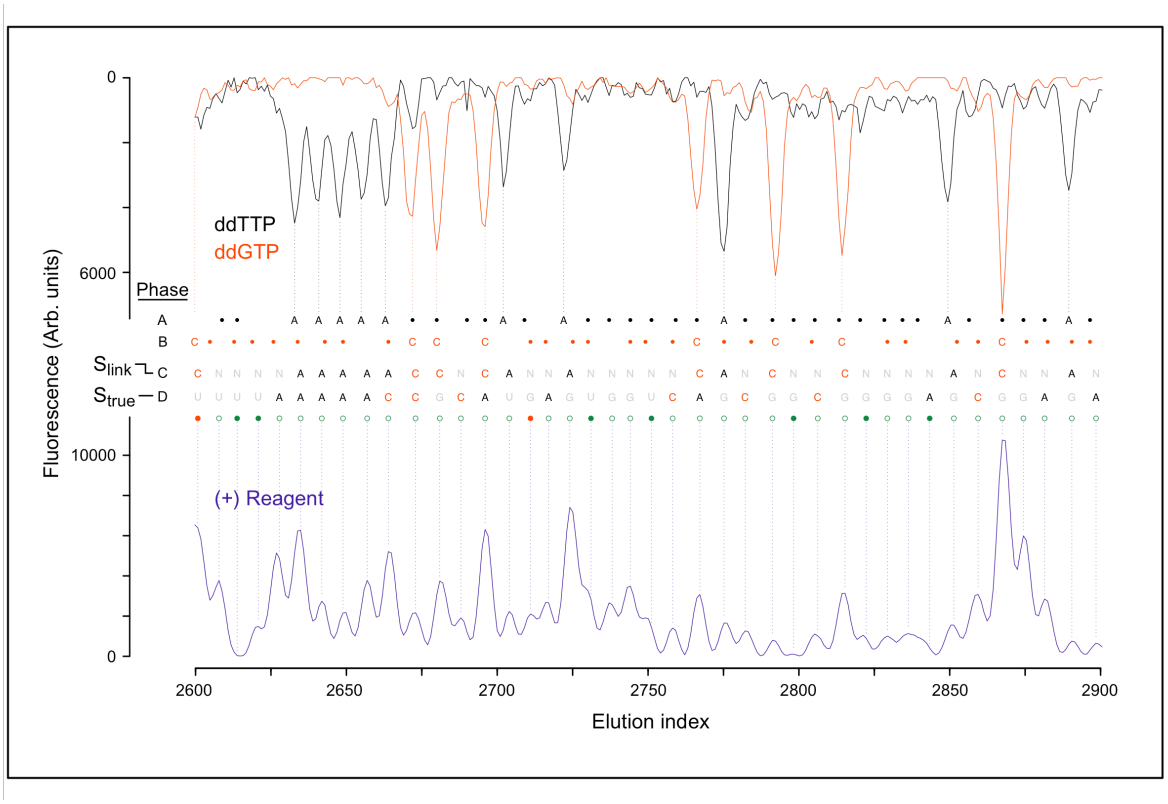

Figure 2.9. Sequence alignment. Electropherogram showing the (+) reagent (bottom) and the ddNTP (top) channels. Sequencing channels have been inverted to facilitate visualization of peak synchronization with the reagent channel. Reagent peaks assigned in the alignment step (**Figure 2.8**) are highlighted with blue dotted lines. Results of the four phases of sequence assignment are plotted together with their respective spectra. (A,B) Detection of peaks corresponding to the first and second sequencing channels, respectively. Peaks not accepted as valid sequencing positions are shown with black and red filled circles. (C) Assignment of input sequence to the identified sequencing peaks. (D) Complete alignment of the input RNA sequence. This alignment is offset by one nucleotide to reflect that dideoxy sequencing fragments are 1 nucleotide longer then the cDNA fragments that identify 2'-*O*-adduct sites.

(3) Editing of the alignment. After the initial alignment, mismatches in the alignment of S_{link} with S_{true} may be observed. The alignment can be edited by manually adding or deleting peaks in the channels in the Modify panel. When adding a (+) reagent peak, adding a corresponding (–) reagent peak is often

necessary. Generating a correct alignment by adding and deleting peaks is an iterative process.

(4) Whole Channel Gaussian Peak Fitting. Once the alignment is correct, the intensity of each peak in the (+) and (–) reagent channels is quantified by fitting a Gaussian curve to each peak in the entire channel (Eqn. 2). The three variables that characterize each peak are the peak area (A_i) and the center and width of peak *i* (μ_i and σ_i, respectively). Since the center of the peak, μ_i, was determined during the peak finding phase, this equation has two unknowns for each peak: area, A_i, and peak width, σ_i. ShapeFinder implements an exhaustive search algorithm to optimize A and σ for each peak. The search algorithm is executed several times, with each iteration refining the search space for A_i and σ_i.

Initial estimates of A_i and σ_i for a given peak are calculated from a local three-peak Gaussian fit of the target peak and the neighboring peaks on each side. Initial values of A_i are taken from $\gamma_i/2 \le A_i \le 10\gamma_i$, where γ_i is the amplitude of the peak fluorescence intensity. σ_i estimates are taken from the range $0.8 \le \sigma_i \le 4.5$. The estimation is repeated for 16 iterations where the sample space of A is adjusted each round to $A_{i,best} - 0.5A_{i,best} \le A_i \le A_{i,best} + 0.5A_{i,best}$, where $A_{i,best}$ is the area calculation from the previous round which best fit the data. The next iteration of the search algorithm refines estimates for A_i and σ_i by using a different sample space for σ_i, $0.4\sigma_{med} \le \sigma_i \le \sigma_i + 0.5\sigma_{med}$, where σ_{med} is the peak width median calculated from all σ_i estimated previously. These steps yield good agreement

between the experimental and fit intensities, although the peak area is slightly underestimated (Figure 2.5A).

If the Optimize option is enabled, estimates of A and σ are improved further at the cost of increased processor time. New parameters are estimated by sampling $A_i \leq A_{new} \leq A_i + 10A_i$, and fixing the width, ω, as the minimum σ_i computed thus far; each σ_i is retained as $\sigma_{i,old}$ for the future. As the initial new ω is taken as the minimum σ_i computed so far, the new A estimates are larger to compensate for the smaller ω. In the final phase of the Optimize algorithm, peak widths are improved in two stages. In the first stage, peak widths are optimized by sampling a new σ_i from $\omega \leq \sigma_{i,new} \leq \sigma_{i,old}$, starting with the peak with the worst fit. After each selected peak is optimized, then a new peak with the worst fit is determined. This is repeated n times, where n is equal to 3 times the number of identified peaks. In the second stage, the peak with the worst fit is again determined and peak width is optimized by sampling, $\sigma_i \leq \sigma_{i,new} \leq \sigma_i + 0.1\sigma_{i,old}$, provided $\sigma_i + 0.1\sigma_{i,old} < \sigma_{i,old}$. The new width is saved if it improves the fit, otherwise the old information is retained. Results of this final optimization step are shown in Figure 2.5B. Fitting ~400 nts of the HIV-1 sample data requires ~16 min on a 1.5 GHz Power PC processor versus 3 min with the Optimize option disabled.

The absolute reactivities for all analytical peaks in the trace are then calculated by subtracting the (–) reagent areas from those for the (+) reagent channel. Absolute reactivities as a function of nucleotide position are output in text files. The Input versus Fit File contains the calculated curve fit to the (+) reagent (reagent) and (–) reagent (background) channels. The Integrated Peaks File

contains a tab-delimited spreadsheet of the calculated peak positions, widths, areas, and RMS errors for the (+) reagent (RX) and (–) reagent (BG) channels, as well as their alignment to the target RNA sequence. This file also contains a column where the (–) peak areas are subtracted from their corresponding (+) peak areas to determine absolute hSHAPE reactivity (Figure 2.6C).

Since the lengths of each cDNA fragment in the sequencing channel is 1 nucleotide longer the (+) reagent channel, the sequencing alignment is shifted by one nucleotide such that (+)/(–) reagent reactivity information is attributed to the correct nucleotide position (Figure 2.9, Phase D). Only the Integrated Peaks File reflects this shift; previous processing steps do not account for this offset.

2.5.3 Statistical Analyses.

All statistical analyses were performed using R [44]. For all analyses, if a nucleotide was present in one data set, but absent in the others, the nucleotide was removed from the analysis. Pearson's correlation coefficients were computed for each possible pairing of the 5 HIV-1 data sets, resulting in 10 calculated correlation coefficients per position. One-way ANOVA and Levene's tests were employed for determining mean reactivity differences and differences in reactivity variation among the 5 HIV data sets, respectively.

Chapter 3
Application of Sequence Alignment Algorithm to High-Throughput RNA Structure Analysis

3.1 Abstract

Selective 2'-hydroxyl acylation by primer extension (SHAPE) is a chemical modification technique used in the analysis and prediction of RNA secondary structure. Currently, the ShapeFinder software suite facilitates analysis of SHAPE experiments. After post-processing of the data captured by DNA sequencing equipment, the ShapeFinder tool Align and Integrate is used to identify peak positions, align the peaks to the known RNA sequence, and then finally quantify per nucleotide flexibility information. The alignment step of the tool requires a manual editing by the user as the peak identification step can incorrectly or mis-identify peaks. This can be a very time consuming step for the experimenter and thus we've developed a new alignment algorithm to aid in the automation of the alignment process. The new algorithm is based on the classic global sequence alignment algorithm, resulting in a significant improvement of the overall alignment step and minimal editing by the user, by proposing a more accurate alignment as well as possible errors in peak identification.

3.2 Introduction

Selective 2'-hydroxyl acylation by primer extension or SHAPE is a technique combined with computational analysis methods to predict RNA secondary structures

as shown in Figure 3.1. SHAPE, as described in Chapter 2, is a chemical modification technique which targets all four nucleotide types of an RNA to identify paired and unpaired regions [7, 8, 27]. The output of a SHAPE experiment is a spectrogram collected from DNA sequencing equipment [6]. A signal processing software suite, ShapeFinder (described in Chapter 2), was developed to process the SHAPE spectra in order to identify and quantify per nucleotide reactivity information [9, 10]. The per nucleotide reactivity data is used as a pseudo-free energy constraint in secondary structure prediction algorithms such as RNAstructure [45, 46]. From this, more accurate RNA structures are produced, in a high-throughput manner[6].

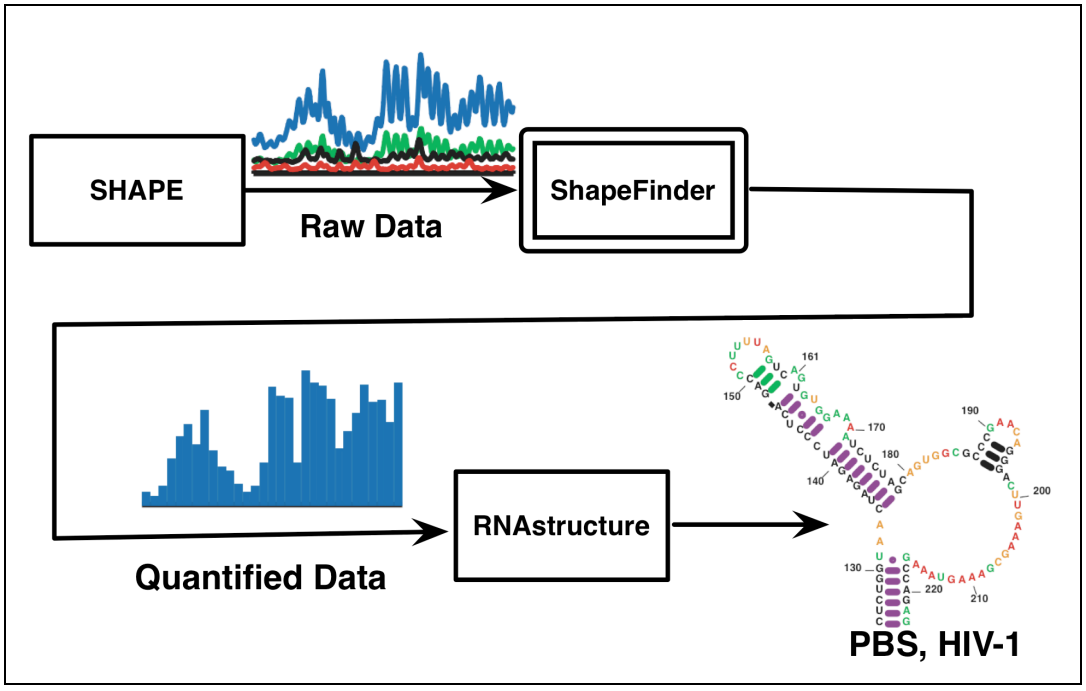

Figure 3.1. Block diagram of steps involved in determining RNA secondary structure using SHAPE. The output of SHAPE is a spectrogram whose signal is then processed by the ShapeFinder tool set. ShapeFinder produces quantified per nucleotide flexibility information. RNAstructe, a secondary structure prediction algorithm, takes the SHAPE data as a pseudo-free energy constraint producing a secondary structure prediction such as this predicted structure of the primer binding site of the HIV-1 genome.

Before quantifying per nucleotide flexibility information, the ShapeFinder algorithm aligns the experiment to the known RNA sequence being analyzed. This step is necessary to ensure that the correct SHAPE modification is attributed to the correct nucleotide. First, the algorithm identifies the experimental sequence. The algorithm then attempts to match the experimental sequence to the RNA sequence by finding the position with the most matching nucleotides. If the experimental sequence has been incorrectly identified by mis-identifying, overlooking or including extra nucleotides, the matching algorithm will incorrectly align the sequences. A manual editing step is then required by the experimenter to manually align the experiment to the RNA sequence.

The experimental sequence is found by identifying peaks in the spectra that correspond to the sequenced nucleotides. If a peak has been missed or incorrectly identified this causes the experimental sequence to be incorrect with either extra or missing nucleotides. The manual editing step used by the experimenter involves, manually adding or deleting identified peaks with a simple click on the computer screen, (Figure 2.2, gray boxes on top of peaks). At present, this is a very time consuming, manual process and detracts from the high-throughput aims of the SHAPE technique. To automate this process, a new technique using a dynamic programming algorithm adapted from the global sequence alignment algorithm is presented. Results are presented showing reduced involvement of an experimenter to manipulate the data, as well as an improvement in accuracy over the previous method.

3.3 Results

The new alignment algorithm was tested on SHAPE traces from two different RNAs. One trace was a relatively noise-free, clean trace. The other contained a great deal of noise. The global alignment was implemented in the programming language Objective-C. It was integrated within the ShapeFinder software architecture as part of the Align and Integrate Tool [10].

The first dataset consisted of relatively clean data captured on a portion of RNA analyzed from the 5' end of the HIV-1 genome (provided by Kevin Wilkinson from the Weeks lab) [6]. This dataset was relatively clean and noise free and contained a strong signal. The new alignment was performed and compared to the old best-matches algorithm. There is an obvious improvement in the alignment. Insertions are denoted with a '-' sign as shown in Figure 3.2a, New. In the old alignment, the 'X' represents both sequencing nucleotides, Figure 3.2a, Old. Since there are peaks in both sequencing lanes in the same position, the old peak detection and alignment algorithm cannot determine which nucleotide to attribute to the peak. However, in the new alignment algorithm, the 'X' is replaced with the aligned nucleotide as determined in the back trace. The new alignment algorithm found the correct starting position of the experimental sequence to the RNA sequence. The results of the new alignment algorithm are very close to the final alignment produced with the older algorithm after the addition and deletion of peaks (data not shown).

The second dataset was from of the third intron of the cytochrome B transcript of the *Saccharomyces cerevisiae* mitochondrial genome. This RNA is composed of approximately 85% A's and U's (provided by Caia Duncan of the Weeks lab) [38].

This dataset was extremely noisy. However, with the new alignment algorithm we see a significant improvement in the alignment, Figure 3.2b. The correct region of the RNA sequence is aligned and exactly matching the start position when comparing with the old alignment after peaks have been added or deleted (data not shown).

For both these datasets the gap penalty for insertions in the experimental sequence was set to 2 and a gap extension penalty was not used.

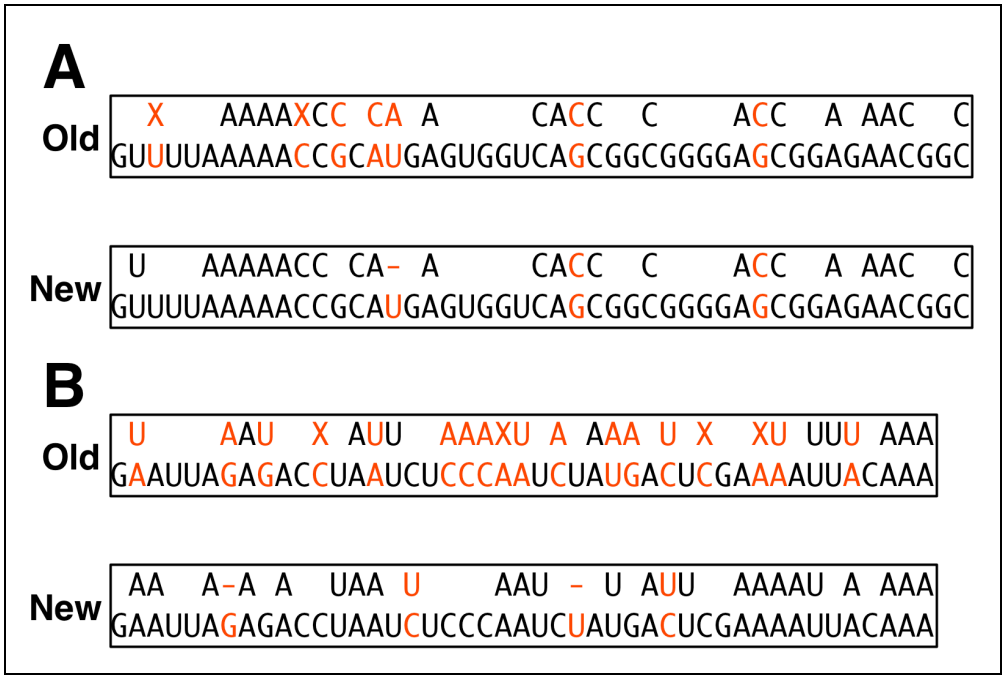

Figure 3.2. This is a portion of the alignment captured from the algorithm. The top sequence alignment is the new global alignment routine. The second sequence alignment is the old best matches algorithm. Red is used to highlight mismatches. A) HIV-1 B) *Saccharomyces cerevisiae*.

3.4 Discussion

The new alignment algorithm correctly identifies the portion of the true RNA sequence to align to the sequenced RNA. This is significant as the experimental RNA sequence is typically a subset of the true RNA sequence as SHAPE typically

can only analyze 300-500 nucleotides in an experiment. The length of the experimental RNA sequence is usually less then the true RNA sequence. Gaps in the alignment were inserted and a visual indication was given in the ShapeFinder software in order to aid the experimenter. However, the interpolation of new peaks remains in the hands of the user who will still have to manually add or remove peaks.

An additional challenge in this implementation was the determination of gap penalties. At the moment the gap penalty for gaps in the true RNA sequence is set very high in order to 'discourage' gaps. Please see Materials and Methods for more information. However, a proper gap penalty for the experimental RNA sequence gaps is still under study. The need for a gap extension penalty has not been ruled out and has been implemented in the code.

There is still some error in the association of the reagent peak intensity to a specific nucleotide. Both algorithms, best matches and the global alignment, are very sensitive to the alignment of peaks between the channels. However, with the new alignment algorithm the time it takes to correct the alignment by adding or deleting peaks is greatly improved. Once the peak detection and peak channel alignment algorithms have been improved, there will be even less user interaction required to process RNA SHAPE data.

Overall, this is viewed as a major improvement over the previous best matches algorithm. When using the older algorithm on noisy data, it was difficult for a user to determine where to start with the process of manually editing the peaks. With this new algorithm, alignment with the correct portion of the true RNA sequence

along with the identified gaps will greatly facilitate and reduce the time involved in manually identifying peaks to add or remove. The association of a reagent peak to the correct nucleotide will provide more accurate nucleotide flexibility information.

3.5 Material and Methods

For the sequence alignment performed in SHAPE experiments, we are comparing the same RNA sequence with itself, Figure 3.3. In a sense, the experimentally determined sequence is a sub-portion of the RNA sequence, contained within. A global alignment algorithm that handles an alignment within a sequence was implemented [47].

A matrix, *F*, is created to calculate the score of the alignment between each position of the experimentally determined sequence and the true RNA Sequence based on the Needleman-Wunsch global alignment algorithm [48]. The recursion relationship is as follows where there are separate gap penalties for each sequence.

$$F(i,j) = \max\begin{cases} F(i-1,j-1) + s(S_i^{true}, S_i^{link}), \\ F(i-1,j) - d, \\ F(i,j-1) - e. \end{cases} \quad (1)$$

The matrix is initialized as in the Smith-Waterman local alignment algorithm [49], where *F(i,0)* and *F(0,j)* are set to 0, where *i = 1, …, n* and *j = 1, …, m*. *F(0,0)* is set to 0. The maximum score, F_{max}, is found on either the bottom or right border of the matrix, i.e., *(i,m)* or *(n,j)*, *i = 1, .., n* and *j = 1,..,m*.

Figure 3.3. An example of a global alignment of a sequence, x, contained within a sequence, y.

Once *Fmax* has been found on one of the edges of the matrix, the trace back begins forming the best alignment of the two sequences in reverse.

The score matrix used in the first equation of the relation in equation 1, is a simple 4x4 matrix construct as follows:

$$\begin{array}{c} \\ NT1 \\ NT2 \\ N \\ X \end{array} \begin{array}{c} \begin{array}{cccc} NT1 & NT2 & N & X \end{array} \\ \begin{bmatrix} 2 & -1 & -1 & 1 \\ -1 & 2 & -1 & 1 \\ -1 & -1 & 2 & -1 \\ 1 & 1 & -1 & 2 \end{bmatrix} \end{array} \quad (2)$$

NT1 and *NT2* represent the sequencing channels in the experiment, i.e, the black and red channels in Figure 2.2, [6, 7]. The choice of nucleotide for the sequencing channels can change from experiment to experiment and are generically represented by *NT1* and *NT2*. With this generic representation, the score matrix will not change from experiment to experiment. The experimental sequence is extracted from the alignment of the reagent peaks to the sequencing channels, Figure 2.2. Peaks not linked to a sequencing peak are generically represented as N in the derived sequence as they are nucleotides in the sequenced RNA. For example, CACNCCCNAANNCNNCCNCCNCAAA… is an example of a derived sequence. In this example, A is *NT1* and C is *NT2*. The current experimental sequence identifying algorithm cannot resolve a peak as either *NT1* or *NT2* when there is a strong peak at

the same position in both *NT1* and *NT2* channels. It is then represented as an 'X'. Since it could be either *NT1* or *NT2* in the sequence, it is considered a match but it is given a lower match score, i.e., *s(NT1,X)* = 1, while *s(NT1,NT1) = 2*.

The RNA sequence is originally extracted from an input file containing the complete RNA sequence. In order to perform the alignment it is converted such that all non-*NT1* and non-*NT2* nucleotides are transformed to N. For example, CACGCGCGAAGU is converted to CACNCNCNAANN, if *NT1*=A and *NT2*=C.

The last portion of the alignment algorithm is the definition of the gap penalties. There are two separate gap penalties for insertions and deletions. The gap penalty for insertions, *e,* in the true RNA sequence is set very high in order to discourage insertions. At present, *e* is set to 50. However, deletions or insertions in experimental sequence will be allowed, as it is an indication of a possible missed peak in the peak identification portion of the algorithm. The gap opening penalty of *d*=2 was used. A gap extension penalty, *f*, is implemented in the algorithm, but unused at this time.

Chapter 4
Influence of Nucleotide Identity on Ribose 2'-hydroxyl Reactivity in RNA

4.1 Abstract[2]

Hydroxyl-selective electrophiles, including N-methylisatoic anhydride (NMIA) and 1-methyl-7-nitroisatoic anhydride (1M7), are broadly useful for RNA structure analysis because they react preferentially with the ribose 2'-OH group at conformationally unconstrained or flexible nucleotides. Each nucleotide in an RNA has the potential to form an adduct with these reagents to yield a comprehensive, nucleotide-resolution, view of RNA structure. However, it is possible that factors other than local structure modulate reactivity. To evaluate the influence of base identity on the intrinsic reactivity of each nucleotide, we analyze NMIA and 1M7 reactivity using four distinct RNAs, under both native and denaturing conditions. We show that guanosine and adenosine residues have identical intrinsic 2'-hydroxyl reactivities at pH 8.0 and are 1.6 and 1.3 times more reactive than uridine and cytidine, respectively. These subtle, but statistically significant, differences do not impact the ability of SHAPE (selective 2'-hydroxyl acylation analyzed by primer extension) based methods to establish an RNA secondary structure or monitor RNA

[2] This work was published in *RNA* (2009) 15(7):1314-21. Reproduced with permission from Cold Spring Harbor Laboratory Press.

folding in solution because base-specific influences are much smaller than the reactivity differences between paired and unpaired nucleotides.

4.2 Introduction

Chemical and enzymatic probing of RNA are critical tools in structural biology and have contributed enormously to our understanding of RNA structure and dynamics and of complex formation with proteins and other ligands. Central to these methods are probes that react with RNA, usually to induce cleavage or modification at flexible, unpaired, or solvent-accessible regions. Ideally, probe reactivity should depend exclusively on RNA structure or solvent accessibility, take place *in vivo* or under physiologically relevant conditions, be independent of nucleotide identity, and not require significant RNA-to-RNA optimization.

SHAPE, or selective 2'-hydroxyl acylation analyzed by primer extension, is well suited for analysis of local nucleotide structure and dynamics because it interrogates all four RNA nucleotides in a single, robust experiment [7]. SHAPE uses hydroxyl-selective electrophiles such as *N*-methylisatoic anhydride (NMIA) and 1-methyl-7-nitroisatoic anhydride (1M7) [7, 11] that react with the 2'-hydroxyl group at conformationally flexible or disordered nucleotides [7, 50] to form a 2'-*O*-ester product (Figure 4.1A). Sites of modification are then identified by primer extension. SHAPE chemistry reports the positions of unpaired or otherwise conformationally unconstrained nucleotides under mild, structure-reinforcing conditions; shows good reactivity towards all four RNA nucleotides on the minute timescale; is suitable for *in vivo* RNA structure analysis; and does not require significant optimization because concurrent reagent hydrolysis makes a separate quench step unnecessary.

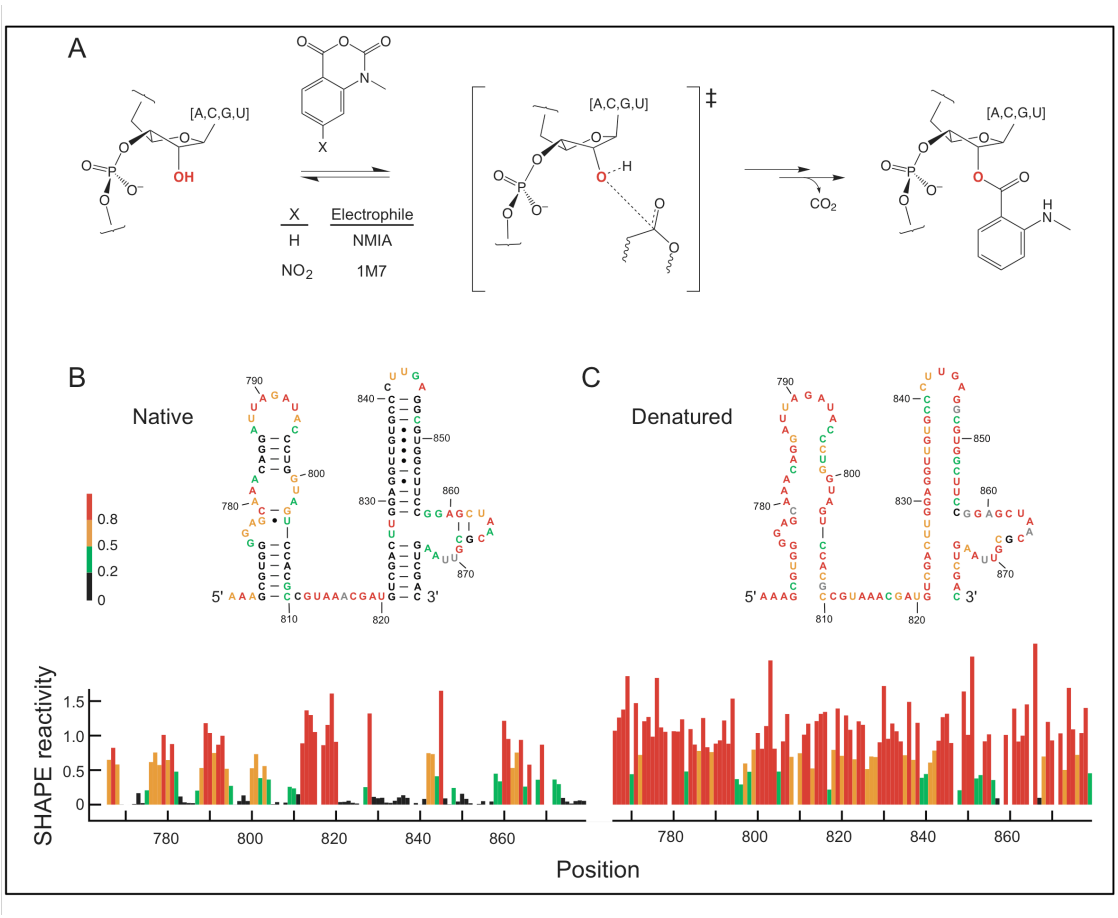

Figure 4.1. Reaction of electrophiles with the 2'-hydroxyl position in RNA. (A) Scheme for SHAPE chemistry. (B,C) Secondary structures and normalized SHAPE reactivities analyzed under native and denaturing conditions. A representative region of 16S rRNA is shown.

Although the overall correlation between local nucleotide flexibility and SHAPE is now well established [7, 11, 50], there are at least three current observations that suggest factors other than RNA structure might influence the reactivity of NMIA or 1M7. First, cytidine residues in flexible regions sometimes have lower SHAPE reactivities than other nucleotides with similar apparent local structures [29, 30, 36]. Second, the pK_a of the ribose 2'-hydroxyl group varies by as much as 0.5 units, as a function of the nucleobase in RNA mono- and dinucleotides [51-53]. Third, the rate of

base-catalyzed in-line cleavage, which is also sensitive to the 2'-hydroxyl pK_a, varies by up to 4-fold as a function of nucleotide sequence [54].

The most general model that explains the strong relationship between SHAPE reactivity and local nucleotide flexibility is that 2'-hydroxyl acylation occurs preferentially at rare conformations that are sampled more frequently at flexible or disordered sites [7]. Any factor that modulates the nucleophilicity of the 2'-hydroxyl group, including potential electronic crosstalk between the RNA base and 2'-hydroxyl, will in turn modulate the yield of 2'-*O*-ester adducts and measured SHAPE reactivity.

In this work, we develop a systematic approach to explore the extent to which nucleotide identity modulates NMIA and 1M7 reactivity with the 2'-hydroxyl in diverse RNAs. Using a bootstrap analysis of variance (ANOVA), we show that all four RNA nucleotides react similarly with both NMIA and 1M7 but that there are small, statistically significant, differences such that purines are slightly more reactive than pyrimidines.

4.3 Results

4.3.1 Strategy.

To assess the influence of base identity on SHAPE reactivity, we analyzed four structurally diverse RNAs. For each RNA, structure was interrogated both under conditions that stabilize native secondary and tertiary folding and also under strongly denaturing conditions. The four RNAs included a transcript corresponding to the 5' end of an HIV-1 genome, the specificity domain of *Bacillus subtilis* RNase P, and portions of the *Escherichia coli* 16S and 23S rRNAs. The HIV-1 transcript includes

the first 976 nucleotides from the 5' end of the genome and contains both a highly structured regulatory region as well as less structured RNA coding regions [6]. The RNase P specificity domain (154 nts) from the thermophilic prokaryote *B. subtilis* forms a compact, well-defined, and highly constrained structure [55]. Finally, we analyzed ~400 nt internal segments from authentic 16*S* and 23*S* rRNAs, isolated from *E. coli* [45]. These diverse RNAs represent a cross section of many typical RNA motifs.

Using these four RNAs, we obtained structural constraints using both the NMIA and 1M7 SHAPE reagents for a total analysis of 5,128 nucleotides. This dataset is sufficiently large to establish rigorously the intrinsic SHAPE reactivities of each RNA nucleotide.

4.3.2 Statistical analysis of intrinsic reactivity in denatured RNA.

To measure the intrinsic reactivity of each of the four nucleotides, we performed SHAPE experiments on all four RNAs using both 1M7 and NMIA under denaturing conditions [20 mM Hepes (pH 8.0) at 90 ˚C]. As expected, nucleotides are consistently more reactive than is observed for the natively folded RNAs (for example, compare Figure 4.1B and C). However, some nucleotides (green and black bars, Figure 4.1C) remained unreactive under denaturing conditions. The existence of these unreactive positions may indicate that base pairs or other structural constraints still form in RNA under our denaturing conditions. In order to make clear conclusions regarding *intrinsic* nucleotide reactivities not confounded by residual RNA structure, we analyzed two subsets of our 2,411 denaturing condition measurements. The first group consists of the entire dataset, which assumes that

the RNA is completely denatured and that low reactivities do reflect unconstrained positions (Figure 4.2). The second group exluded nucleotides that form internal Watson-Crick base pairs in the accepted secondary structure for these RNAs. Our assumption was that nucleotides in internal pairs form the strongest interactions and are therefore the most likely to remain paired under denaturing conditions.

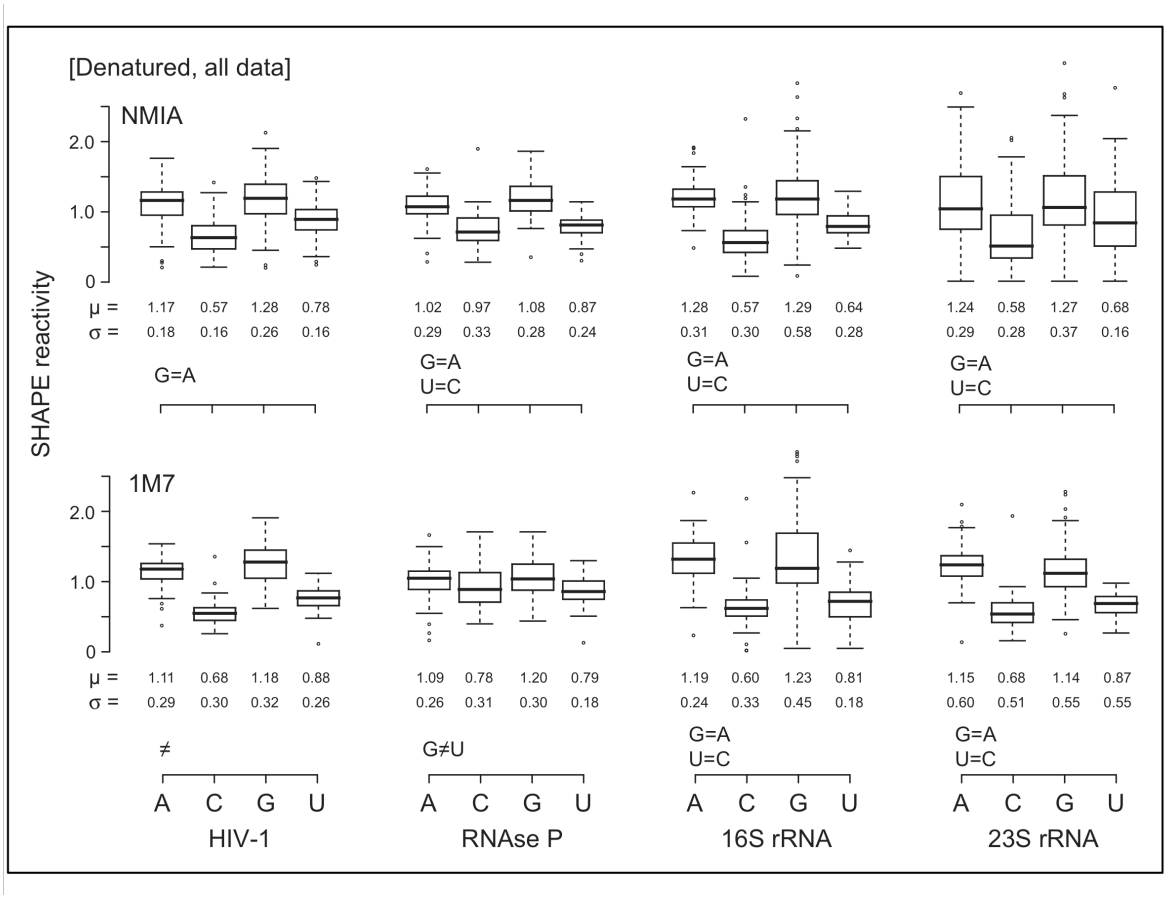

Figure 4.2. Box plot analysis of SHAPE reactivities for the entire denatured RNA dataset. Upper and lower panels illustrate experiments performed with NMIA and 1M7. Equalities at the bottom left of each group emphasize nucleotides showing statistically equivalent reactivities. Boxes outline the middle 50% of each dataset; medians are indicated with bold lines. Whiskers above and below each box give the largest or smallest non-outlier values; outliers are indicated by open circles and are >1.5 times the interquartile range (box).

We report our reactivity data using box plots [56], a convenient way to visualize statistics for large datasets (Figure 4.2). The box spans the central half (from 25% to

75%) of the data and the median is shown with a heavy line. The vertical distance between the upper and lower box edges is termed the interquartile range (IQR). Whiskers in the box plot illustrate 1.5 times the IQR. Values outside this range are commonly taken to be outliers [56] and are shown explicitly as circles. We also report the mean (μ) and standard deviation (σ) for each dataset.

Visual inspection of box plots showing nucleotide reactivities under denaturing conditions suggests that reactivities follow a clear trend, G ≈ A > U > C, for both reagents and all four RNAs (Figure 4.2 and data not shown). We evaluated the statistical significance of these differences using bootstrap analysis of variance (ANOVA) [57] and multiple comparison procedures [58]. The bootstrap ANOVA showed statistically significant ($p<0.05$) differences in nucleotide reactivity for the group consisting of the entire dataset, see Table 4-1.

Table 4-1. Reagent statistics. = indicates react equally; ≠ indicates don't react equally

Reagent	*RNA*	*Homoscedastic*	*ANOVA*	*MCP*
NMIA	HIV	Yes, $p = 0.195$	$p < 0.0001$	G=A
	RNAseP	No, $p = 0.031$	$p < 0.0001$	G=A, U=C
	16S rRNA	No, $p < 0.0001$	$p < 0.0001$	G=A
	23S rRNA	Yes, $p=0.729$	$p < 0.0001$	G=A, U=C
1M7	HIV	No, $p < 0.0001$	$p < 0.0001$	≠
	RNAseP	Yes, $p = 0.063$	$p = 0.0048$	C=A=G,U=C
	16S rRNA	No, $p < 0.0001$	$p < 0.0001$	G=A,U=C
	23S rRNA	No, $p = 0.0009$	$p < 0.0001$	G=A,U=C

The results of the multiple comparison procedures (summarized at lower left of analysis, Figure 4.2 and in Table 4-1) emphasize that the two purine residues, guanosine and adenosine, had statistically equivalent reactivities and that the

pyrimidines also reacted similarly. A few datasets, notably RNase P with 1M7, showed a small departure from this trend because all nucleotides showed equal reactivity.

The most critical result is how robust the overall reactivity trends are. The overall reactivity trend and mean reactivities are nearly identical for both the NMIA and 1M7 electrophiles, independent of whether internal pairs are removed (Figure 4.2 and data not shown). These results emphasize that we are measuring intrinsic nucleotide reactivities that are not influenced by residual structure under denaturing conditions.

4.3.3 Analysis of native state RNA.

To determine whether nucleotide identity influences SHAPE reactivity in fully folded RNAs, we also analyzed nucleotide reactivities for the four RNAs equilibrated under conditions that enforce native structure (in the presence of Mg^{2+} at 37 ˚C) prior to reaction with NMIA or 1M7.

For the folded native RNAs, reactivities vary dramatically because nucleotides experience many different local nucleotide environments. We therefore separated nucleotide reactivities for each RNA into four groups (i) unpaired; (ii) paired, but adjacent to unpaired or non-canonically paired nucleotides (termed externally paired); (iii) paired, and adjacent to other canonically paired nucleotides (internally paired); and (iv) non-canonically paired. As expected, unpaired nucleotides exhibited the highest mean (~0.66 SHAPE units) and standard deviation (~0.60) in reactivities, followed by the external pairs. Internal base pairs showed the lowest mean reactivity (~0.09) and reactivity variability (σ ~0.17) (Figure 4.3). Non-canonically paired

nucleotides had idiosyncratic profiles, characterized by variability in measurement means and standard deviations for each RNA.

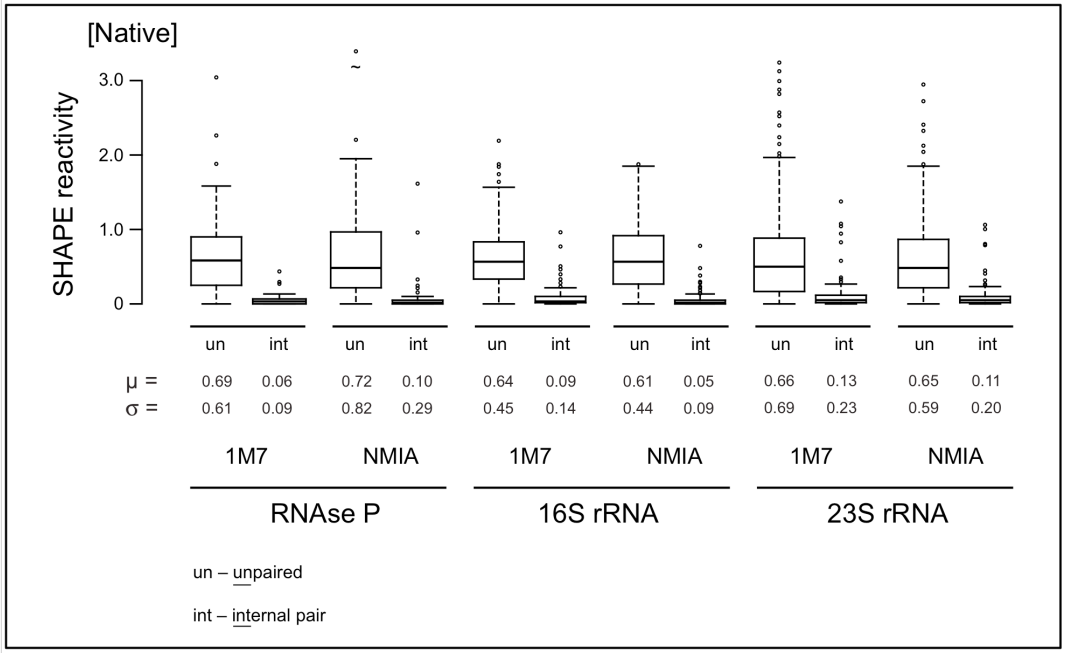

Figure 4.3. Differential reactivity of unpaired (un) and internally (int) paired nucleotides towards NMIA and 1M7. Paired nucleotides react within a tighter range and have a smaller mean reactivity than do unpaired nucleotides.

Multiple comparison procedures confirmed that unpaired and internal pairs have very different reactivities (Figure 4.3) consistent with the basic model that SHAPE measures nucleotide flexibility. The statistical analysis also confirmed that SHAPE measures small differences in nucleotide environment because most datasets also exhibited statistically significant differences between unpaired and externally paired nucleotides.

We also assessed intrinsic nucleotide reactivities for unpaired nucleotides (group i) in the context of the folded RNAs (Figure 4.4). These datasets were relatively small because fully unpaired nucleotides comprised only 32–48% of each RNA. Statistical differences between nucleotide reactivities are less pronounced

than those for the denatured RNAs and it was not possible to detect base-specific trends at the $p<0.05$ level. However, qualitative inspection of the reactivity datasets for unpaired nucleotides under native conditions reveals the same overall trend as observed under denaturing conditions. Adenosine and guanosine nucleotides are generally, but not always, more reactive than cytidine and uridine; cytidine residues were consistently the least reactive in all datasets (Figure 4.4).

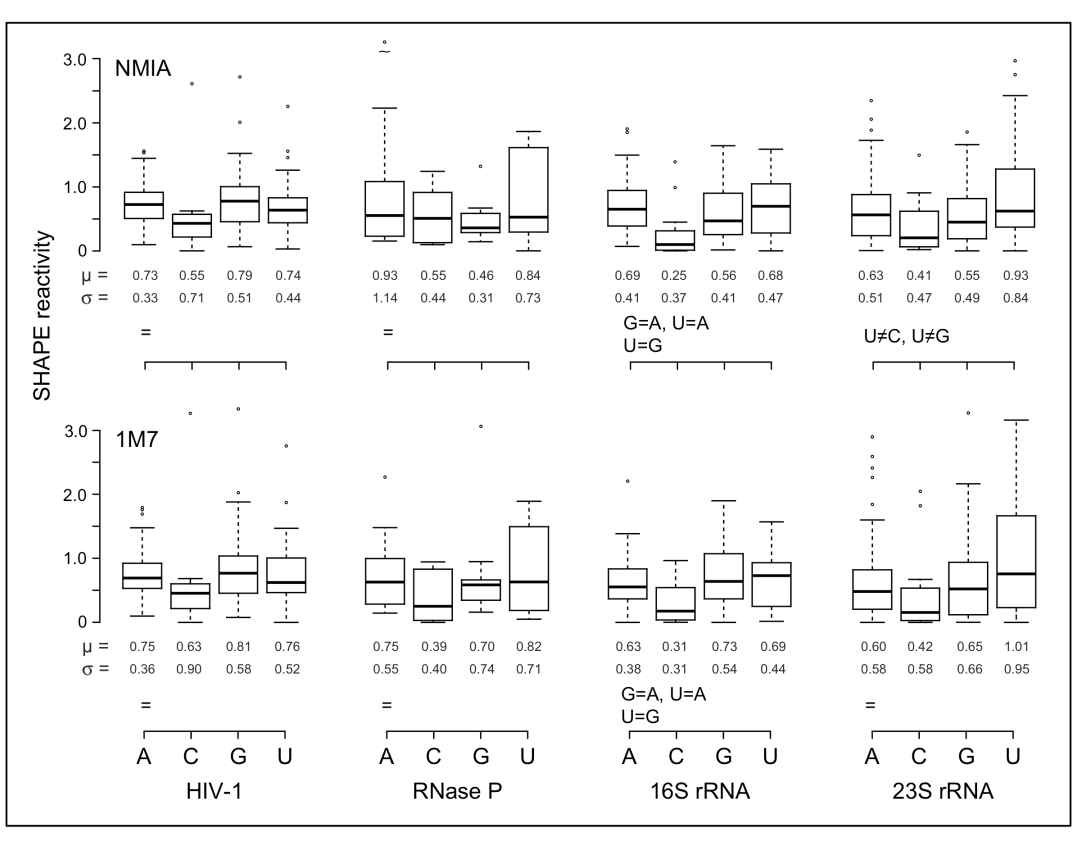

Figure 4.4. Box plots of nucleotides that are single stranded in natively folded RNAs. Reactivities towards NMIA and 1M7 are shown in the upper and lower panels, respectively. Statistical equalities are indicated at the lower left of each plot.

4.4 Discussion

4.4.1 SHAPE chemistry is much more sensitive to RNA structure than to nucleotide identity.

SHAPE chemistry reports local nucleotide flexibility and disorder via reaction at the ribose 2'-hydroxyl position, consistent with a mechanism in which flexible nucleotides are better able to sample relatively rare, but highly reactive, conformations (Figure 4.1A). We establish, based on an analysis of over 5000 nucleotide reactivity measurements, that the nucleobase has only a weak influence on SHAPE reactivity (summarized in box, **Figure 4.5**), and that this influence is small relative to the contribution of local RNA structure. Two lines of evidence support these conclusions.

First, stably paired nucleotides consistently react to a lower yield and exhibit a smaller range in reactivities than do unpaired nucleotides (Figure 4.3). The average ratio of mean SHAPE reactivity for unpaired and internally paired nucleotides is 7.3. Thus, SHAPE chemistry has a strong positive predictive ability for differentiating paired and unpaired nucleotides.

Second, no consistent, statistically enforceable trend in reactivity as a function of nucleotide identity can be discerned based on unpaired nucleotides in natively structured RNAs. Thus, the weak structural interactions that occur in the single stranded regions of typical folded RNAs have a larger effect on reactivity than any intrinsic difference imposed by nucleotide identity (Figure 4.4). If nucleotide reactivity were more important, a consistent trend would be obvious in the single stranded regions of native-state RNAs.

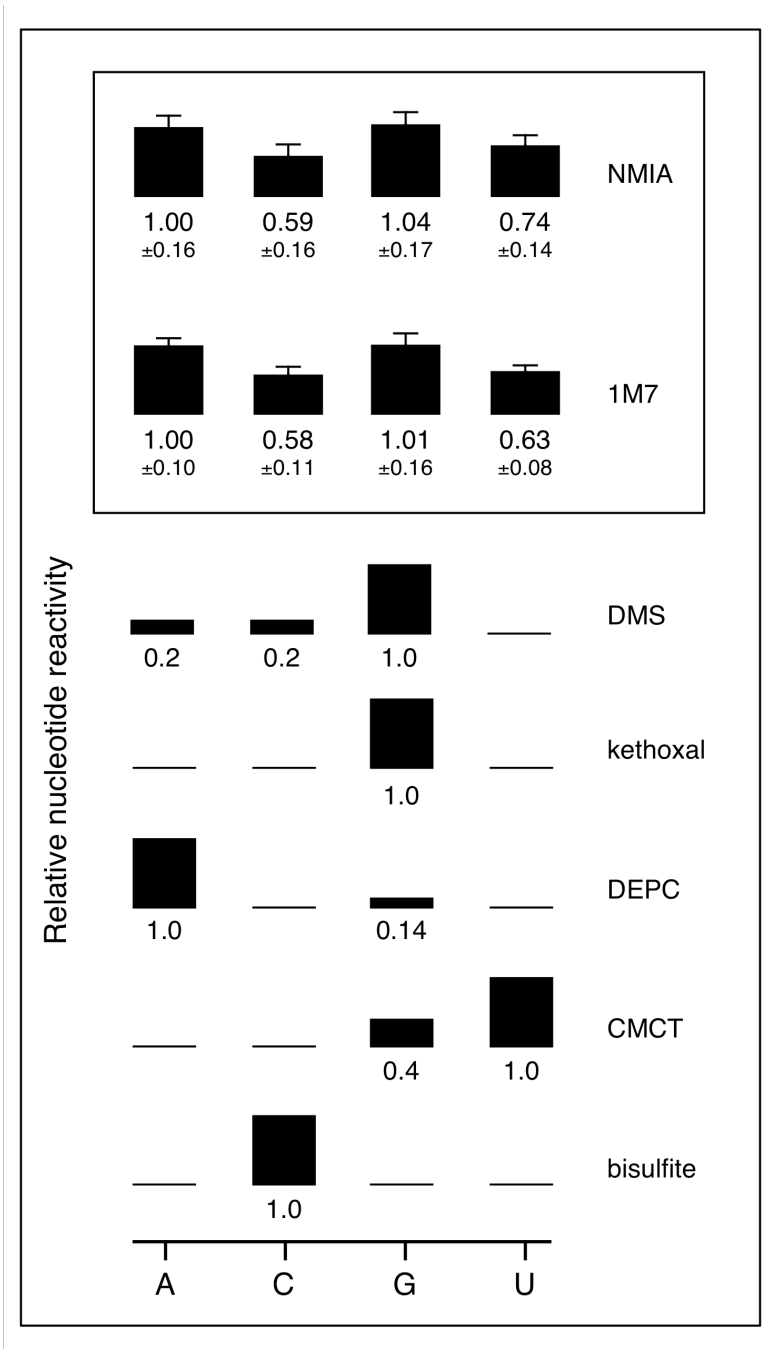

Figure 4.5. Nucleotide-specific reactivities for NMIA and 1M7. Reactivities for NMIA and 1M7 are reported as the mean plus an error term (root mean square of the coefficient of variation). These values are compared with other reagents that form stable covalent adducts with RNA. Numerical estimates for the relative nucleotide-specific reactivity of each reagent were obtained from: DMS [59], bisulfite and kethoxal [60], DEPC [61], CMCT [62].

Intrinsic nucleotide reactivities are also independent of the choice of SHAPE electrophile. In general, we recommend the use of 1M7 for routine analysis of RNA structures at equilibrium. 1M7 is less sensitive to variations in ion concentrations [11] and to the possible contributions of slow conformational dynamics at specific RNA nucleotides [63].

4.4.2 Accurate prediction of RNA structure based on experimental chemical modification information requires a pseudo-free energy change approach.

SHAPE reactivities strongly discriminate between paired and unpaired nucleotides, regardless of nucleotide identity (Figure 4.3). However, it is also evident that some unpaired nucleotides have reactivities of zero (lower whiskers on un-labeled box plots, Figure 4.3), while some internally paired nucleotides have moderate to high reactivities (upper whiskers on int-labeled box plots, Figure 4.3). There is no clear demarcation between paired and unpaired nucleotides. Thus, any RNA structure prediction algorithm that imposes a hard cutoff between paired and unpaired nucleotides is guaranteed to introduce inaccuracies in a structure calculation. Although we have performed our statistical analysis using SHAPE information, this result likely applies even more strongly to conventional chemical probes for RNA structure like base-selective chemical probes and nucleolytic enzymes [11, 46, 64].

SHAPE reactivities can be converted into pairing probabilities or pseudo-free energy change terms and used to constrain existing thermodynamic models for RNA folding to determine RNA secondary structures with accuracies often greater than 95% correct [45]. This pseudo-energy approach is highly tolerant of experimental and

other errors because each reactivity measurement provides an energetic bias to the final structure calculation, but does not impose an abrupt cut-off [6, 45].

We investigated whether correcting the observed SHAPE reactivities by the difference in intrinsic reactivities for the four nucleotides changes or improves secondary structure prediction for the HIV-1, RNase P or ribosomal RNAs when energy biases are introduced as a base pair stacking term [45, 65]. The predicted structures are essentially identical and exhibit the same overall topology as those reported previously [6, 11, 45, 55]. Small differences were observed at the ends of some helices and at multi-helix junctions. We infer that nucleotide-to-nucleotide variation in reactivity is not a significant source of error in SHAPE-directed RNA structure prediction.

4.4.3 Comparison of NMIA and 1M7 reactivities to other reagents used to map RNA structure.

We developed general parameters for the intrinsic nucleotide reactivities for NMIA and 1M7 (**Figure 4.5**). These relative reactivities were calculated from experiments performed under denaturing conditions (Figure 4.2) and therefore represent the largest possible difference in reactivity between the four RNA nucleotides. The key result is that NMIA and 1M7 react broadly with all four nucleotides. The intrinsic reactivity order is A ≈ G > U > C. However, the maximal bias between purines and pyrimidines is less than 2-fold. This even reactivity stands in stark contrast to traditional structure-selective reagents that also react to form stable covalent adducts with RNA (**Figure 4.5**). The local nucleotide environment can also be probed in degradative reactions using the lead(II) ion [66] and base catalyzed in-line probing [67]. Both of these approaches share with SHAPE the

feature that they react broadly with the four RNA nucleotides. Structure-specific cleavage with lead(II) is influenced by ion affinity towards specific RNA structures [68] and in-line probing reactivities vary by up to 4-fold as a function of base identity [54].

The physical basis for the differences in SHAPE reactivities between purine and pyrimidine residues is not completely clear. The pK_a of the 2'-hydroxyl in dinucleotides is higher for the pyrimidines (at ~12.8) than for adenosine (~12.5) or guanosine (~12.7) [52]. Since 2'-*O*-adduct formation involves loss of this proton, the trend in pK_a values offer a partial explanation for differing intrinsic reactivities. In some RNAs, a subset of cytidine residues, drawn as single stranded, have low SHAPE reactivities [29, 30, 36]. Given that the intrinsic reactivity of this nucleotide is, at most, only 2-fold different from the other nucleotides, we postulate that these cytidines may participate in a locally constraining interaction that remains to be fully characterized.

In sum, the differential effects of base identity on 2'-hydroxyl reactivity, while statistically significant, are small compared to the larger influence of local RNA structure on 2'-hydroxyl reactivity. That the least structured nucleotides show the largest reactivities strongly supports the initial model [7, 27] that SHAPE reactivity is primarily governed by local nucleotide flexibility.

4.5 Materials and Methods

4.5.1 SHAPE on HIV-1, RNase P, and ribosomal RNAs.

The general procedures for SHAPE analysis of the four RNAs studied in this work – the HIV-1, RNase P, and 16S and 23S rRNAs – were described previously. The HIV-1 RNA is a transcript of the 5'-most 975 nucleotides from the NL4-3 strain

[6]; the RNase P specificity domain RNA is imbedded in 5' and 3' structure cassette sequences [7, 11]; and 16S and 23S rRNA are authentic ribosomal RNA, purified from *E. coli* [45]. Accepted secondary structures for these RNAs were taken from the following sources [6, 55, 69]. Experiments performed under denaturing conditions employed 20 mM Hepes (pH 8.0) at 90 ˚C for 4 min. Native-state modification experiments were performed at 37 ˚C in 50 mM Hepes (pH 8.0), 200 mM potassium acetate and 5 mM $MgCl_2$, except for the RNase P RNA which were performed in 100 mM Hepes (pH 8.0), 100 mM NaCl, and 10 mM $MgCl_2$. RNAs were generally allowed to equilibrate in buffer for 30 minutes prior to addition of reagent. RNAs were initially incubated in a buffer containing 10/9 of these concentrations and reactions were initiated by addition of 1/10 volume of DMSO containing 1M7 or NMIA. No-reaction controls contained neat DMSO. The 10× NMIA stock concentration was 130 mM for all RNAs. The 1M7 stock (10×) concentration was 30 mM for 16S and 23S rRNAs, 100 mM for RNase P, and 50 mM for the HIV-1 RNA. Following modification, RNAs were recovered by ethanol precipitation and resuspended in 1/2× TE [5 mM Tris (pH 8.0), 0.5 mM EDTA]. Each SHAPE reaction product [(+) and (–) reagent and 1 or 2 sequencing ladders] was analyzed using primers labeled with distinct fluorophores as described [6, 11, 45].

4.5.2 SHAPE data processing.

Primer extension products were resolved on an ABI 3130 capillary electrophoresis DNA sequencer using custom fluorescence spectral calibration. Runs typically yielded more than 400 nts of structural information for the long RNAs. Raw electropherograms were analyzed using the signal processing framework in

ShapeFinder [10]; areas were calculated for all peaks in the (+) and (–) reagent channels by Gaussian peak fitting. Absolute SHAPE reactivities were calculated by subtracting the (–) peak areas from the (+) peak areas. Positions exhibiting high background were discarded; reactivities that were slightly less than zero were reset to zero. SHAPE datasets were scaled such that a generic reactive nucleotide has an intensity of 1.0 and an unreactive nucleotide is 0. Reactivities for the native datasets were therefore normalized by dividing by the average of the 10% of the most reactive positions, after discarding points with reactivities greater than the third quartile plus 1.5 times the interquartile range. For denatured data, we assume that nearly all nucleotides are unconstrained: peak areas were normalized by dividing each data point by the average reactivity of all peaks.

4.5.3 Statistical analysis of intrinsic nucleotide reactivities.

A standard one-way Analysis of Variance (ANOVA) relies on assumptions of independence, normality, and homogeneous variances between groups (termed homoscedasticity) [41]. Quantile-Quantile (Q-Q) plots and the randomized Levene's test [70] indicated that SHAPE data are not normally distributed and can be heteroscedastic. Therefore, a bootstrap ANOVA [57], which does not rely on assumptions of normality, was used to assess if the observed reactivities reflect intrinsic reactivity differences or chance. In the bootstrap ANOVA, reactivities, independent of nucleotide identity, were randomly sampled from the measured SHAPE reactivities and used to re-form the original group sizes. Resampling was performed 15,000 times and an F statistic calculated for each iteration. The proportion of F values that are greater than or equal to the F statistic for the original

data is reported as a p-value; p-values less than 0.05 indicate that differences observed between groups in the original data are statistically significant. When the bootstrap ANOVA found statistically significant differences in reactivity as a function of nucleotide type, randomized multiple comparison procedures for homoscedastic and heteroscedastic nucleotide groups were performed to identify statistically equivalent or unequal groups [58]. For **Figure 4.5**, nucleotide reactivities are reported as the mean and the root mean square coefficient of variation. Statistical analyses were performed using R [44].

4.5.4 Structure prediction.

SHAPE-directed structure determination was performed using RNAstructure [46] using SHAPE reactivity information as a pseudo-free energy change term [45].

Chapter 5
Agent-based model of the dynamics of phenotype switching in *Bacillus subtilis*

5.1 Abstract[3]

Competence is a DNA uptake phenotype in *Bacillus subtilis* expressed by bistable switching in genetically identical bacteria populations. The nature of "noise" in the stochastic emergence of the competence phenotype has been difficult to examine directly by lab experimentation. Computational modeling is an alternative approach that can be used to examine the sources of noise in the gene regulatory network responsible for the emergence of competence. We developed a multi-scale agent-based model to examine competence switching both at the molecular and population levels in *B. subtilis*. At the bottom level, our model consists of an agent-based model of the intracellular molecules involved in regulating the competence network to produce the phenotype switching behavior of a single cell. Multiple cell models are then incorporated into a cell culture model, with feedback between the multi-cellular model and intracellular model driven by extracellular nutrient concentrations and cell density conditions. From the model, we observe that (i) spatio-temporal randomness in initial agent (molecule) placement may explain the stochasticity of the competence switch, (ii) molecular concentrations of key

[3] This chapter has been submitted for publication with the authors Suzy M. Vasa and Morgan C. Giddings.

competence molecules inherited through cell division influences the emergence of competence to provide a type of epigenetic heritability and (iii) the dilution effect of cell division upon protein concentrations may explain why competence only emerges in stationary phase.

5.2 Introduction

The notion that a bacterial genotype is the sole driver of its phenotype has been challenged by the soil bacterium *B. subtilis.* This organism exhibits a DNA uptake phenotype in a fraction of genetically identical bacteria during stationary phase growth. This competence phenotype is driven by changes in gene and protein expression states rather than changes in genotype. This phenotype appears to be a cell survival strategy to obtain new genetic information, repair DNA or obtain DNA as food [17]. Competence emergence is correlated with high cell density and nutrient limiting conditions [17], where approximately 10-20% of a *B. subtilis* population will express the competence phenotype under these conditions [17].

Through such examples observed in bacteria, evidence has accumulated that genetically identical cells can express differential phenotypes without having differential genotypes. Phenotype switching can be due to stochastic intracellular molecular interactions, or due to environmental inputs like changing nutrients or cell density [15, 71-73]. Such mechanisms are called “bistable” switches, implying that there are two stable genetic regulatory states, between which the cell can switch with appropriate input. Bistable switching is thought to be a mechanism by which bacteria may rapidly adapt to changing environments without the need for slower and perhaps more costly genetic change [74].

Bistability can be viewed as a simple two state system consisting of an on state (phenotype expressed) and an off state (no phenotype) (Figure 5.1a). The two states are the 'stable states', whereas in-between states are unstable and transient. The ability of a bistable switch to transition to an on-state is governed in part by intracellular noise, i.e., random variations in biochemical reactions [18, 75, 76]. There is also a growing body of evidence demonstrating that transitions from one bacterial cell phenotype to another are often governed by regulatory auto-feedback loops (e.g. Figure 5.1b) [16, 71]. In the simple feedback loop illustrated, the system is comprised of a protein product of geneX, which binds at its own promoter site to enhance expression. The two stable states of the system are OFF, where little or no protein is present, hence there is little or no protein being produced, and ON, where there is enough of the protein present to bind at its promoter and prompt further expression of itself. While the ON state would theoretically drive towards infinite expression, it is usually kept in check by mechanisms of protein degradation, co-factors that regulate gene expression, and others.

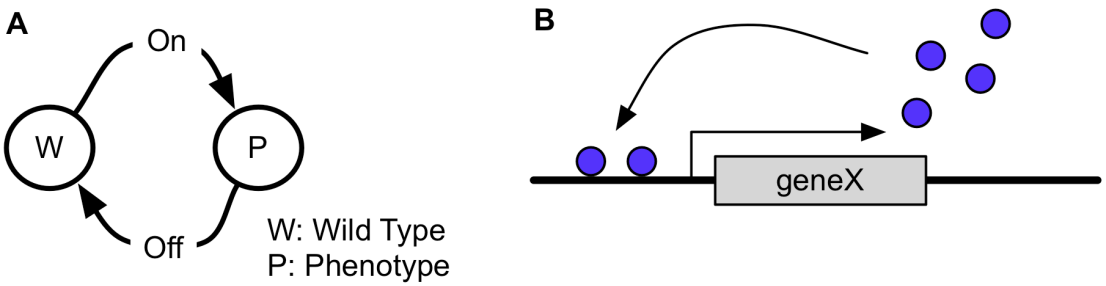

Figure 5.1. Bistable switching in bacteria. A) switching between wild type and a phenotype whether or not a bistable switch has been activated. B) bistability in positive feedback gene regulation. The gene product acts at the promoter to enable its own transcription.

The competence phenotype in *B. subtilis* is driven by bistable expression of the *comK* transcription factor [18], which is involved in a feedback loop that regulates its own expression and controls the downstream phenotype. Accumulation of ComK protein (the ON state) enables downstream transcription of the DNA transport genes [77] that lead to the observed competence phenotype. Since *comK* is a "switch" that drives a key phenotypic state, its expression is tightly regulated [78] at both the transcript and protein levels (Figure 5.2). Yet despite the strong regulation of this circuit, the semi-random appearance of the competence phenotype under stationary growth conditions indicates stochasticity in the regulatory network involving ComK and ancillary actors like ComS. This stochasticity may derive from the very low expression levels of ComK, where a change in presence or absence of only a few ComK molecules may lead to a phenotypic state change. Varying spatial arrangements and temporal interactions of low-abundance molecules like ComK are suspected to be a factor in the variable expression of the phenotype.

However, the stochastic nature of bistable switching in the *B. subtilis* competence mechanism has been difficult to tease apart. Mathematical modeling has been employed in an attempt to understand the nature of this process [18, 79-81]. These models address the stochastic nature of competence by modeling noise in the system with varying degrees of specificity using pre-defined noise terms as well as the Gillespie stochastic modeling algorithm [82]. While such models have led to further insights, there have been several challenges. First, it is difficult to represent concentrations of a molecule like ComK with a bulk rate equation since it is present in such small quantities. For example, if only one ComK copy is present and

another one is then produced, the concentration is immediately doubled in a discrete (non-linear) fashion. Second, the noise terms to drive state changes are added explicitly to the model rather than derived from the model's inherent structure. This has the limitation that one must define those noise terms correctly ahead of time, and may not yield insight into the underlying nature and/or source of noise in the real cell.

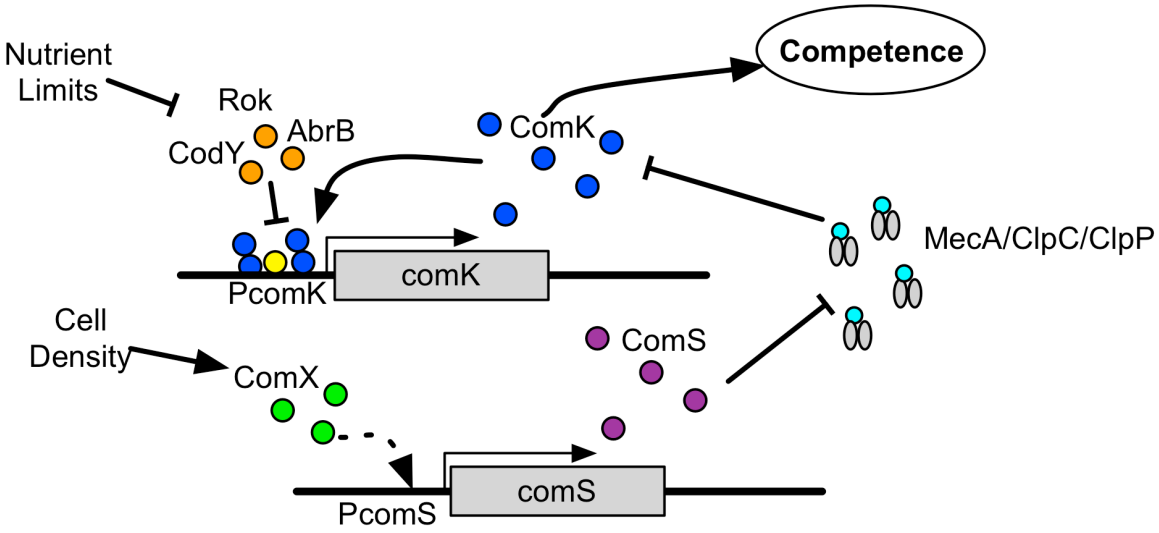

Figure 5.2. Regulation of competence by a bistable circuit centered on ComK. Significant quantities of ComK will activate downstream competence genes. However, ComK expression is regulated pre-transcriptionally by repressor proteins and regulated post-translationally by degradation by the MecA/ClpC/ClpP protease complex. ComS competes with ComK to bind to the MecA adapter protein. Increased ComS production will then decrease degradation of the ComK protein. The transcription regulator DegU is shown in yellow.

To address these challenges and gain further insight into the molecular mechanisms of phenotype switching, we applied a novel agent-based modeling technique (ABM) to study bistable switching in *B. subtilis*. In ABM, the entities of interest in the model are defined as 'agents'. Each agent is an independent, information carrying, decision-making entity representing an individual molecular

entity such as a protein, DNA, or RNA. In the ABM, agents interact with each other and their environment based on a set of well-defined, biologically related rules. Through the myriad of interactions that occur over time, larger patterns of behavior emerge that are not readily predictable from the individual rules, a phenomenon called 'emergence' [2]. Observing the emergent, aggregate behaviors of a model as it responds to adjustment of rules governing individual agents provides a powerful hypothesis-testing mechanism for exploring phenomena such as bistable switching in *B. subtilis*.

For the past two decades, ABMs have been applied to problems in biology mainly within the field of ecology [83]. Here it is often referred to as either individual-based modeling or pattern-oriented modeling [83, 84]. Recently, ABMs have been used to model problems in molecular biology such as biofilm development [85-89], the transmission dynamics of antibiotic resistance [90] and antibiotic resistance mechanisms in *Staphylococcus aureus* [91].

In our ABM, we defined autonomous agents that mirrored key components of the competence switch. Agents representing key proteins, DNAs and RNAs were implemented with rules governing their behavior according to known biological behaviors including stochastic (Brownian) motion, binding, transcription or translation. Since the modeling technique is inherently stochastic due to the initial random placement of agents mirroring the random location of proteins in a cell, we run repeated simulations to determine how initial conditions drive stochastic emergent behaviors like competence.

Here we report our examination of several attributes of the *B. subtilis* competence switch with the ABM, including: how spatial arrangements of the proteins involved in competence can drive the system to distinct end states, the pattern of epigenetic heritability of the competence phenotype, and the effects of nutrient limitation and intracellular signaling on the competence switch in populations of *B. subtilis*. The resulting bottom-up, multi-scale ABM of the *B. subtilis* competence phenotype switching consists of (i) a single cell model, modeling the intracellular interactions of molecules in the gene expression pathway that leads to the transition to the competence state and (ii) a cell culture model consisting of many single cell models interacting with extracellular molecules that influence the competence phenotype, such as nutrients and cell population density. We make the model source code fully available for exploration at http://bioinfo.med.unc.edu, for use with the Open Source modeling platform Repast Simphony [92].

5.3 Results

5.3.1 Intracellular competence models

First, we developed a 3-D intracellular agent based model representing the key molecular players in competence switching in *B. subtilis,* in order to study the mechanisms driving competence. This allowed us to study the sources of the variability in gene expression (noise) that determine switching to and from the competent state. While noise is thought to be a key driver of the phenotype transition to competence, its source and nature has remained ill defined. Using its direct, bottom-up representation of the molecules involved in competence switching,

the ABM allowed us to directly examine the sources of noise in molecular positioning and stochastic interaction.

In the model, we explicitly represent proteins, genes, transcripts, and ribosomes as agents that move and interact in a simulated cellular environment (Figure 5.2). Agent movement follows Brownian motion and is simulated by a random walk implementation (Figure 5.4c and Figure 5.7). When agents bump into other agents during a simulation, they interact stochastically according to their defined behaviors. Agent behaviors (rules) and binding partners are shown in Table 5-3 and discussed in the Methods section. We model the processes of transcription and translation leading from gene to protein, including agents that represent genes, transcripts and ribosomes. By specifying the rules for each agent according to its known properties and interactions, we can see how individual behaviors at the molecular level led to system-wide emergent behaviors like the competent state.

Random transcription of *comK* is a key factor in the build up of large amounts of ComK to trigger transcription of the competence genes [18, 93]. ComK is degraded by a protease complex to keep its levels in check [94]. Another protein, ComS, competes with ComK for degradation [95]. High levels of ComS act to sequester the protease complex to allow further transcription of *comK* [94]. As such, the simulations monitored the concentrations of the ComK and ComS protein agents over time. Time is represented as discrete updates to the state of all agents in the model. A time step is completed when all agent rules have been executed in a random order.

Models were initialized with the agents placed randomly within the simulated 3D grid environment, and were run repeatedly to assess outcomes given different

initial configurations. Emergent competence behavior was monitored by quantifying the overall populations of molecules, particularly ComK and ComS, the former which drives the cell's competence state.

5.3.2 The impact of random spatio-temporal agent arrangement on competence outcome

In an experiment to determine whether the spatial-temporal arrangement of molecules contributes to the determination of the competence state, intracellular model simulations were run with identical parameters and molecular population sizes. The only difference in each run of the simulation was the random initial spatial placement of the agents. Figure 5.3 shows the ComK and ComS population sizes through time in two such intracellular simulations (note the differing y-axis scales). Notably the two simulations diverged significantly in expression of both the modeled ComK and ComS proteins, with Figure 5.3B representing a system being driven to the competence state and Figure 5.3A representing the more frequent case of a system that remains non-competent. We repeatedly observed this phenomenon, with initial parameters and population sizes being identical, yet simulations resulting in significantly different outcomes. Given the parameter settings, the over-production of ComK was observed in approximately 3-5 out of 100 simulations in a given run, as illustrated in Figure 5.3b. It is important to note that these models were run without explicit simulation of starvation conditions that cause a higher density of competence emergence, as shown in the multi-scale, multi-cellular model below.

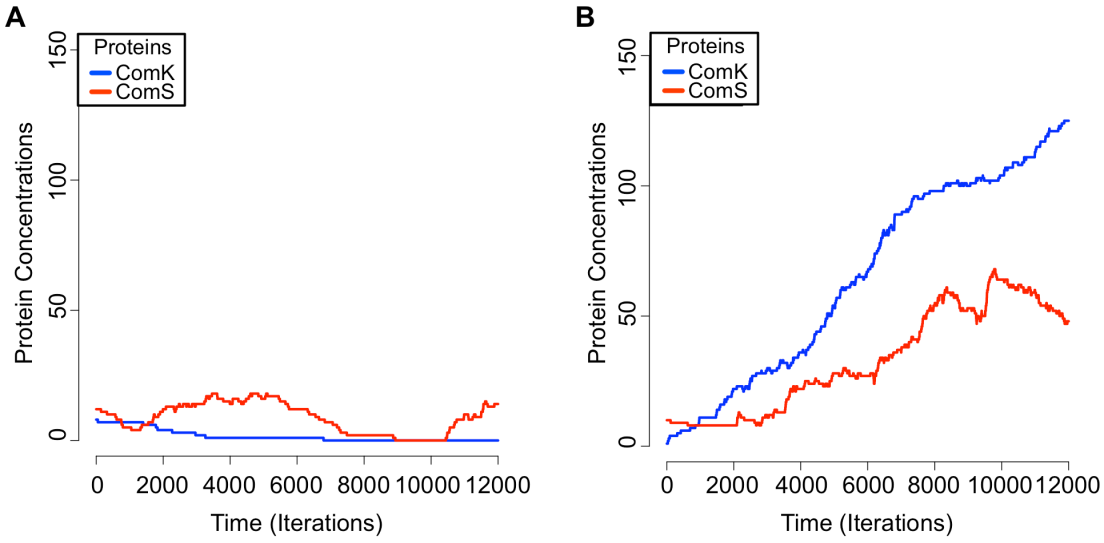

Figure 5.3. Intracellular model where model starts with the same initial concentrations but agents are placed randomly in the environment. A) ComS production exceeds ComK. B) ComK production exceeds ComS.

In a careful study of individual simulation runs, we noted that a chance encounter of a single ComK mRNA transcript with a ribosome before it was degraded would produce ComK protein. This agrees with prior *in vivo* results indicating that an increased probability of competence switching can be driven by fluctuation of the ComK protein by a few molecules [18]. On average, only one mRNA was observed at any given time in the model (data not shown), so cell fate was determined not by its quantity as much as its location and chance encounters with a Ribosome.

These results provide evidence that a random, spatial arrangement of molecules is an important contribution to the variable *comK* gene expression (noise) that drives the competence phenotype. While noise was known to play a key role in bistable switching [16], most prior top-down bulk-rate computational models introduced noise terms that had to be set *a priori*. In our ABM, noise was a natural

outcome of random agent positions and agent interactions. While we cannot assert that random molecular positioning is the exclusive source of the intracellular "noise" that drives competence in *B. subtilis*, the simulation indicated that the random placement of molecules in a cell is perhaps sufficient to drive the random switching seen in *B. subtilis* populations.

5.3.3 Multi-scale, Multi-cellular simulations of nutrient limitation effects on competence

It has been previously shown that nutrient limitation and cell culture density affect the propensity of *B. subtilis* cells to enter the competent state [17]. Our goal was to leverage the intracellular model into a multi-scale model of both intra- and inter-cellular interactions, to examine the cellular population level effects on competence. The model consists of two layers--an intracellular layer and an extracellular layer. The first layer consists of the intracellular model previously described (Within-Cell Model, Figure 5.4). The second layer consists of cell agents representing the whole cell's interaction with extracellular environmental factors such as nutrients and the quorum sensing pheromones (Culture Model, Figure 5.4). Thus, the model is a multi-scale ABM consisting of an overall ABM of agent ABMs running within it.

To model nutrients and quorum sensing pheromones in the extracellular environment, diffusion equation layers were used (Nutrient and Peptide Layers, Figure 5.4). One of the layers represents the local concentration of the ComX pheromone, an intracellular signaling molecule that is involved in quorum sensing and regulation of the competence circuit (Figure 5.2). The Cell agents produce and consume ComX throughout the simulation. Consumption of ComX by a Cell agent

decreases its concentration at the culture level, while resulting in the creation of a new ComX agent in the intracellular ABM. Likewise, the ComX peptide is produced at a constant intracellular rate [96] and is transferred stochastically to the extracellular environment for uptake by other cells (details in Materials and Methods). As concentrations of cell agents grow, extracellular ComX concentrations increase, thereby increasing the likelihood of ComX uptake by other cells.

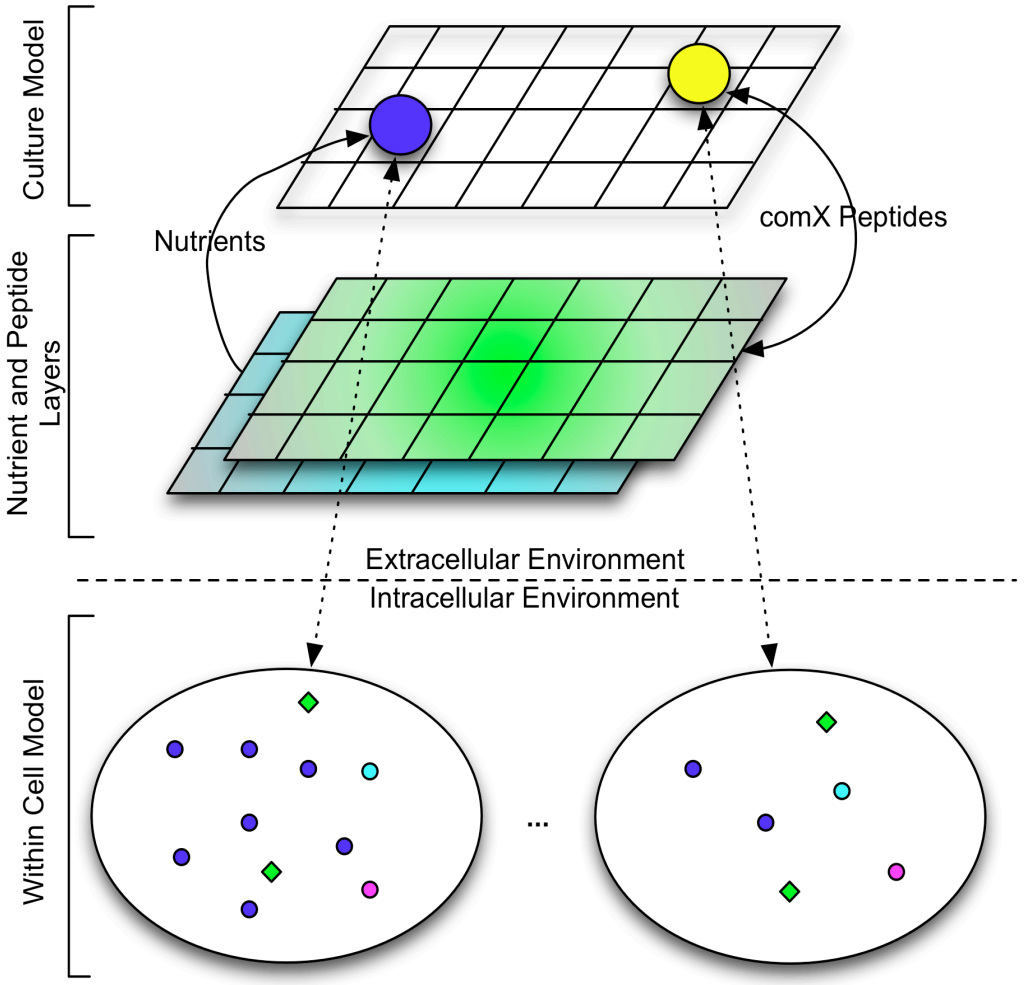

Figure 5.4. The multi-scale agent based model of competence, representing both the intracellular pathways (bottom) and the multicellular environment (top). There are two layers that represent the ComX quorum sensing pheromone and nutrient concentrations.

A second layer represents consumable nutrients required for cell growth and division in the model. Cell agents at the culture level consume nutrients from the nutrient layer, depleting the quantity available in the nutrient layer. The consumption of a nutrient molecule is input to the cell growth equation, based on the Logistic Map equation, that governs growth and division (Methods and Materials). The cellular agents could divide if sufficient growth has occurred according to the equation. As the cell agents grow and divide, daughter agents are placed at a randomly determined adjacent location to the parent. If there is not a free adjacent location, agents are "shoved" to the side to make room for the new cell agent. When nutrients become depleted at a Cell Agent's location, the agent will move in the direction of an increasing nutrient gradient if present, or move randomly otherwise, simulating chemotaxis. If insufficient nutrients are present, a Cell Agent's probability of death is increased. In addition, these starvation conditions are transferred to the intracellular model by reducing the number of agents that repress *comK* and *comS* transcription, Figure 5.2. In this manner, extracellular environmental conditions influence the intracellular conditions, and intracellular conditions feed back upon the environment and other agents within it.

Like the intracellular model, the multi-scale model agents are placed randomly in a 2-D grid environment. However, in this case the initial concentrations of ComK, ComS, and ComX agents for the intracellular models are randomly determined within pre-defined threshold levels (Materials and Methods). There are no ComK mRNA agents placed at the start of a model run, but these agents can be generated via transcription during a simulation.

Since individual simulations would often result in distinct outcomes, we ran the model repeatedly to obtain average statistics for competence-switching behavior. Out of 6 simulations, the model typically reached an average of 867 cells. We necessarily limited the available "plate size" and nutrient concentration for culture growth to limit the computing to feasible time spans. For each of the cell agents, a complete intracellular model was running, which meant that a full simulation running on a fast desktop computer may take three weeks or more to complete. Improved parallelism may reduce run times in the future.

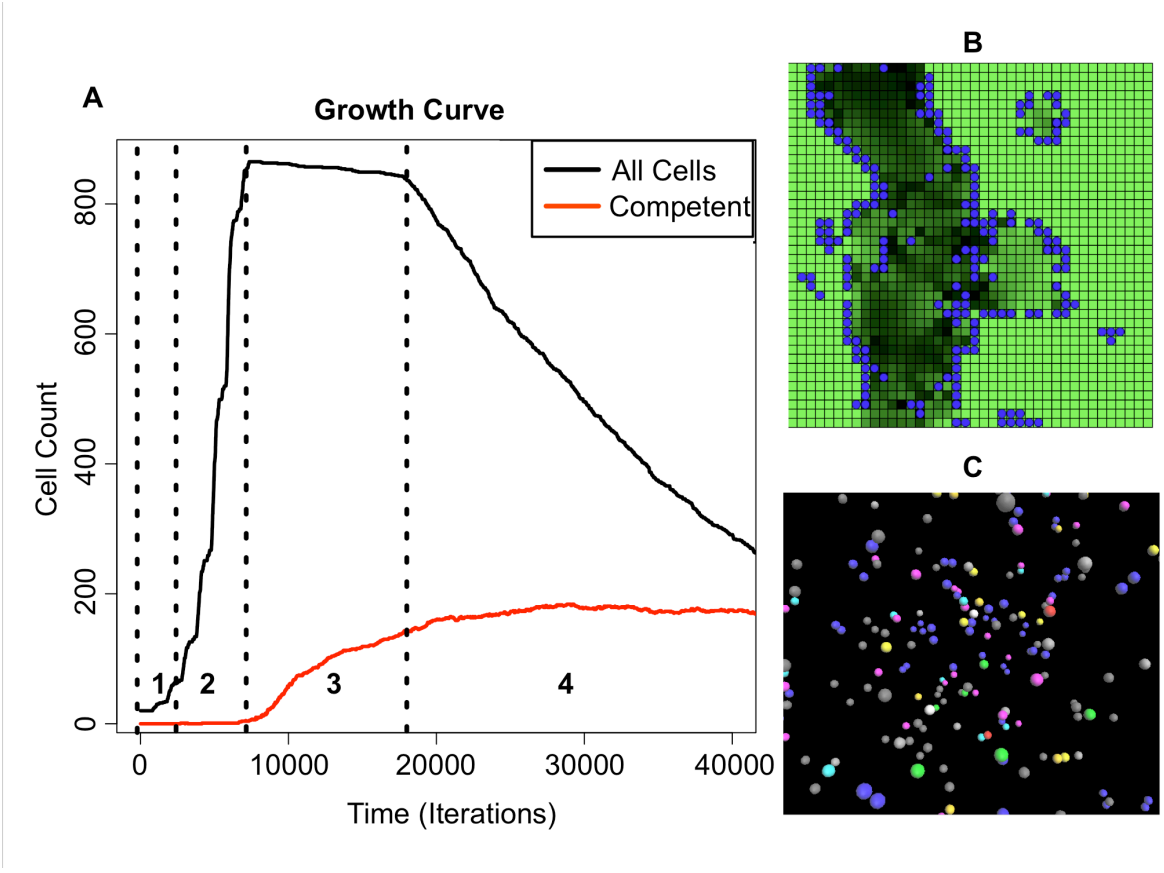

Figure 5.5. A) Growth curve of modeled cell culture. 1-lag phase, 2-exponential growth phase, 3-stationary phase and 4-death phase. B) 2-D agent-based model showing nutrient (green) consumption by cell models (blue) early in the growth phase of the culture. C) A view of the intra-cellular ABM.

Figure 5.5a shows the growth curve for an example multi-scale simulation, with the resulting count of competent cells as they switched to the competence phenotype. In an example simulation, the model was run with an initial seed of 20 randomly placed cell agents that grew to a maximum of about 865 cell agents, with 160 exhibiting the competence phenotype (18.5%) by the end of stationary phase (Figure 5.5a). Execution halted within the death phase of the growth curve after approximately 43,000 iterations. Figure 5.5b shows a snapshot of the simulation, with the blue dots representing cell agents running an independent intracellular ABM. The nutrient gradient is represented in green, with the lighter shade indicating a full nutrient concentration and transitions to darker green indicating nutrient depletion.

The model resulted in classical bacterial growth curves with the standard phases of bacterial culture: lag phase, exponential growth phase, stationary phase, and death phase (Figure 5.5a). It has been shown *in vitro* that competence begins to emerge in abundance during stationary growth phase [17]. As shown in Figure 5.5a, the model demonstrated similar emergence of competence during stationary phase, even though there was no explicit programming instructing it to do so. After cell division ceased due to nutrient limitation, ComK transcripts and protein accumulation increased the likelihood of competence transition. Out of 6 simulations, the intracellular *B. subtilis* ABM demonstrated the emergence of 16.3% (on average) competent cells by the end of stationary phase. This compares with *in vitro* experiments showing 10-20% competence emergence [17].

5.3.4 Modeling the epigenetic heritability of competence

Following from the result showing that the initial spatial distribution of molecules may affect the outcome of the intracellular model, we postulated that some level of epigenetic heritability might exist in the competence switching. The rationale is that if a parent cell has an increased concentration of ComK compared to average, it might be expected that as the cell divides, the resulting progeny may also have elevated ComK levels. This would be dependent in part upon the spatial distribution of the proteins before division, where in some cases, one of the progeny may have received more than their "fair share" of one or more competence-related molecules. Once the competent state is reached, cell division ceases. However, the precursor to competence - accumulation of ComK - may lead to an earlier switch than would occur by random chance alone. There was some evidence for this type of heritability in the biological system presented in Veening et al [97].

We used the multi-scale model to examine the heritability of competence. Each of the 20 initial intracellular models was initialized with concentrations of ComK, ComS, ComX, and mRNAs determined by a random draw from a uniform distribution from within a defined threshold (Materials and Methods). Cells would grow and divide, with division resulting in a random partitioning of its molecular contents among the progeny. All cells were tracked throughout the simulation, and progeny recorded as the model progressed from a minimum of 20 agents to a maximum of 857 agents, with 171 showing competence at simulation termination.

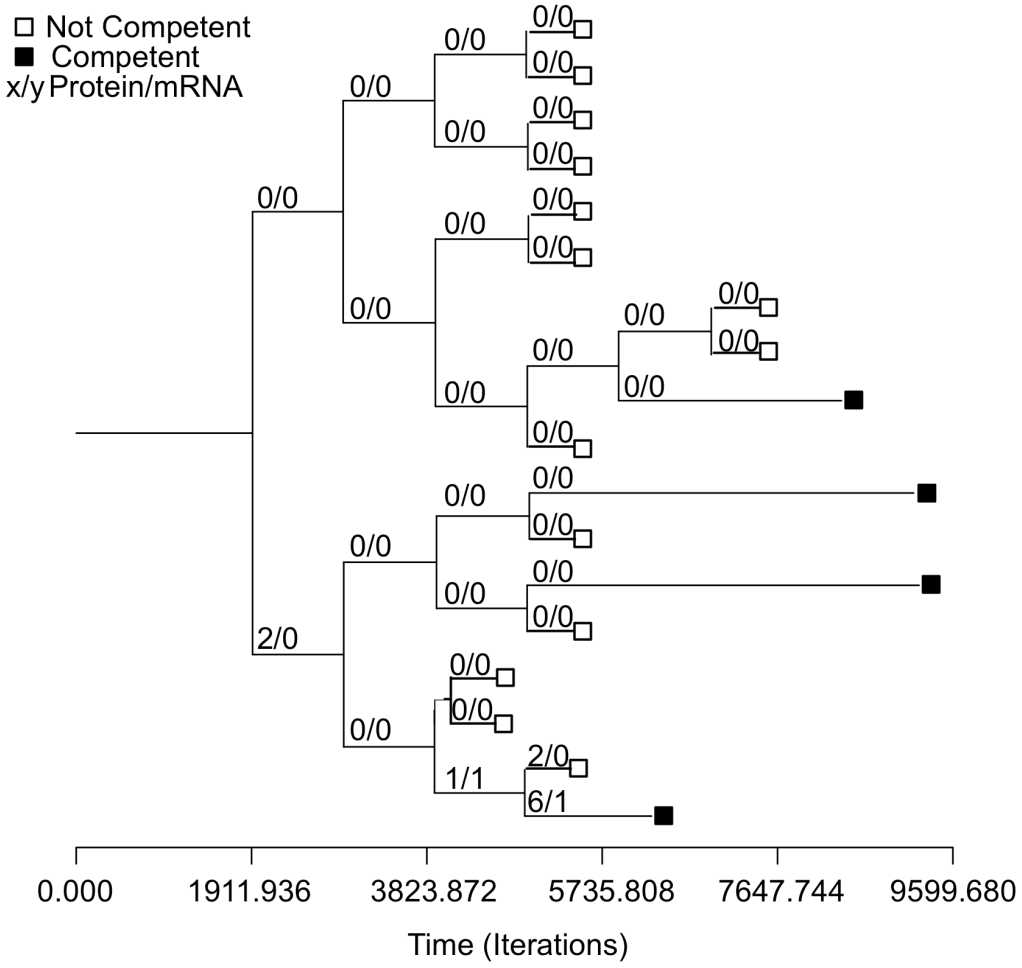

Figure 5.6. By following the life cycle of one cell and its progeny, one can see a pattern of inheritance of ComK transcripts and proteins. At the time of division, concentrations of ComK transcripts and proteins are divided between daughter cells. Protein and mRNA levels are noted at each division. Filled squares indicate the daughter cells, which eventually exhibited the competence phenotype. Branch lengths of non-competent cells (unfilled squares) were artificially lengthened to make room for the labels, otherwise they would be of zero length.

Figure 5.6 shows the lineage following from a single Cell Agent, with ComK protein and mRNA levels denoted at each division. The horizontal axis corresponds to the number of iterations of both the intracellular and extracellular models. As

shown in the bottom branch, daughter cells that initially inherited non-zero ComK levels exhibited the competence phenotype significantly earlier then cells which inherited minimal or no ComK. The concentration of the ComK mRNA never exceeded one per cell, and ComK concentrations ranged from 1 to 8 molecules throughout the exponential growth phase of the simulation. Of the 18 progeny in the lineage shown, four eventually switched to the competent state, with the first transition being along the lineage that had elevated levels.

The model therefore revealed an important feature of how growth stage regulates competence by diluting the mRNA and proteins essential to switching to the competent state. Excepting the lineage that switched to the competent state early due to inheriting elevated levels of the ComK protein, the remaining competent cells only became so after nutrient limitation inhibited cell division and Repressor agents long enough for sufficient concentration of ComK agents to accumulate. This explains why competence in *B. subtilis* typically emerges only once stationary phase is reached and nutrients are limited, due to build up of elements like ComK in the competence gene regulatory network.

5.4 Discussion

Our interest in phenotype switching has derived from studies of anti-microbial tolerant bacterial phenotypes[98]. These are non-inherited phenotypes that confer tolerance towards many known antibiotic drugs. In bacterial populations, cells exhibiting tolerant phenotypes exist as a small fraction of the population, with the quantity determined in part by growth conditions. For example, stationary phase growth (nutrient limitation) induces an increase in the fraction of phenotypically drug

tolerant cells [99]. However, at the present time little is known about the mechanisms underlying antimicrobial tolerance in bacteria. So, to model and explore aspects of phenotypic “bistable” switching, we turned to competence switching in *B. subtilis*, where the key molecular players have already been uncovered.

Our *B. subtilis* agent-based model was built from the bottom-up by specifying rules for individual molecules and their interactions. Properties like the multi-cell growth curve and the competence phenotype were not defined *a priori,* but instead arose naturally from the model as emergent phenomena. For example, the growth curve for the multi-cell model was a natural consequence of basic assumptions about cell division and nutrient uptake. Yet it showed resulting growth curves exhibiting the same features that real bacterial growth curves do: a lag phase, an exponential phase, a stationary phase and death phase. There was no curve fitting involved because this growth curve naturally arose.

More significantly, there was nothing explicitly inserted into the model that indicated that competence should occur in any particular growth phase. The observed fact that cells only turned competent in bulk once stationary phase was entered, was a consequence of the model's assumptions about basic molecular and cellular behaviors. The model revealed a basic but important facet of the emergence of competence: that dilution of competence-determining molecules during cell division acts to regulate the emergence of competence.

The model also showed a specific relationship between the phenotypic (emergent) outcomes of competence with the spatial arrangement of competence-determining molecules. The random spatial division of these molecules into two

subsets during cell division had a strong effect on the determination of competence in the model. And, the random initial spatial arrangement of molecules had a direct effect on whether an individual cell became competent or not. The biological system likely shares similar properties, where the apparent randomness of the competence transitions is derived directly from the randomness of spatial distribution in competence-related molecules. Noise in a biological system is typically thought of as events which explain variability in gene expression. The model reveals that random spatial arrangement of competence-determining molecules may be a major contributor to noise. In the model, when initial concentrations of agents were held constant, a small portion of the executions of the Cell ABM model showed the emergence of the competence phenotype. It appears that spatial interactions, not molecular quantities or agent rule execution probabilities, had the most effect on the emergence of the competence phenotype (e.g. Figure 5.3).

This highlights the power of bottom-up modeling, where only very basic facts or hypotheses about how individual molecules behave are designed into the model, and all the 'complex' behaviors that occur are emergent properties of the system. An emergent property is exemplified by a bacterial growth curve. One can tease apart the workings of an individual cell, and never see a growth curve. It is only by studying how whole populations of cells behave in a shared environment that one might observe the phenomenon of a growth curve which has distinct phases. A primary distinction between the agent-based approach and bulk or top-down modeling is that we do not try to model the growth curve *a priori* with an equation. It

is a consequence of the basic assumptions of molecular and individual cellular behaviors and interactions.

The quantities of molecular agents modeled are a small fraction of the molecules in the live system. Yet the model displayed many of the same properties as the real system, showing how robust the competence mechanism is. We have found similar results in a separate effort modeling chemotaxis in *E. coli* with an ABM, where preserving biological ratios of molecules was more important than preserving absolute quantities, in order to produce biologically realistic results (Miller *et al.,* manuscript in review for *PLOS One*). In this competence model, the robustness is illustrated from the fact that when we incorporate molecular agents in the intracellular model in the estimated proportions that are thought to occur biologically, the model derives competent cells as 10-20% of the population, much in line with biological results.

The cell division trees that monitored concentration levels of ComK displayed an obvious pattern of inheritance. Cells that inherited higher concentrations of ComK transcripts and proteins from the parent cell tended to pass on larger quantities to their children than other cells. Despite the dilution occurring from cell division, cells that inherited ComK transcripts and/or proteins tended to switch to competence more quickly then cells that did not inherit initial concentrations of ComK. On average, it took approximately 3695±1335 iterations for a cell to switch to competence from the last time it divided. The large variability in the length of time to switch to competence appears due to the variability of molecular inheritance. In

this sense, the quantities of molecules inherited in this pathway are acting in an epigenetic fashion upon subsequent phenotypic outcomes.

It is intriguing that the regulation of competence emergence in stationary phase could be tied directly to molecular quantities and the dilution effect of cellular division. During exponential growth phase, not enough time would pass between cell divisions to allow ComK to build up to sufficient levels, so competence does not occur. While we can't know whether this is the complete explanation for how competence is limited to stationary phase in *B. subtilis* cultures, it seems like a sufficient explanation. Usually we think of regulatory mechanisms as being the direct up- or down-regulation of one or more genes, proteins, or post-translational modifications. Yet the model showed that the indirect effect of molecular dilution tied to cell division or lack thereof was sufficient to result in competence emergence during stationary phase. No master regulator was needed. We now wonder how many other cellular mechanisms might work in similar ways, without any master regulators, but simply due to emergent features like the concentration of critical molecules?

While continuous mathematical models of ordinary or partial differential equations model the average behavior of a system (from the top-down), the ABM models the discrete behavior of the components of the system (from the bottom up). Thus, a strength of ABMs is their innate ability to model such spatial dynamics and spatial heterogeneity in a population, which these results indicate as being important in bistable switching. The model showed that a molecular-biological system can be readily translated into an agent-based model where proteins, RNAs and other

molecules become agents; where agents move by mimicking the erratic random movement of soluble biological elements; and where protein interaction networks and metabolic pathways can then be defined by agent rules and interaction probability thresholds. As a result, we gained further insight into bistable switching in *B. subtilis* as a stochastic, spatially oriented process, with several key emergent features deriving from only basic assumptions about individual molecular interactions.

This discrete model of the competence phenotype provides a readily comprehensible view into each cell's behavior and provides the ability to monitor the variation of molecular concentrations involved in regulating competence. The resulting model is biologically intuitive, with ready translation from biological facts or hypotheses into the model and back, without the need to be translated into a system of equations. We suspect that in addition to being a useful tool for biological research, such models will be increasingly used as educational tools in the future because they are straightforward to build, visualize, and comprehend.

5.5 Material and Methods

5.5.1 Modeling environment and overview

All ABMs were developed using Repast Simphony [92]. Repast Simphony is a Java based, open-source ABM framework to facilitate model development.

The model shown in Figure 5.4 is a multi-scale ABM consisting of a cell culture model comprised of bacteria cell agents. Each cell agent is also an ABM simulating the intracellular competence regulatory network focused on the regulation and production of ComK and ComS proteins, as illustrated in Figure 5.2.

Agents represent proteins, mRNAs, promoter sites, repressors and proteases critical to the regulation and emergence of competence. The intracellular ABM is also represented as an agent in the Culture ABM. At simulation start, agents are placed in a grid-like environment. The grid environment simulates either the interior of a cell or the extracellular environment. However, it is a discrete environment where an agent will occupy a single cell in the grid and can move to adjacent cell locations. Agents can only occupy one cell in the grid one at a time, and each cell can hold only one agent. Hence, if a cell is occupied then an agent will move to another location. Agents interact with the grid environment or other agents by stochastically executing rules. Rules are defined by how an agent will move and other agents it is allowed to interact with.

In addition to the two levels of the model, nutrients and the ComX peptide diffusion are mathematically modeled using diffusion equations adapted to the ABM environment. These are modeled using continuous equations rather than as agents due to their high concentrations, and hence their continuously variable nature. Concentrations are monitored in each cell of the grid environment and diffusion to adjacent cells is calculated by the equations described below.

5.5.2 Rules and Agents

In the model, agent movement and interactions with other agents are defined by rules. Rule execution is stochastic and subject to meeting a probability threshold after a random draw from the Uniform distribution (see Parameter Estimation).

For example, if two molecules have been shown to bind biologically with a high affinity, then their interaction probability will be high. In cases where the

literature was not specific enough, probabilities were estimated. For example, Hamoen et al [100] reported that ComK may bind another ComK to form a homodimer. This is represented in the model using an interaction probability ρ (Table 5-1), with the following rule executed whenever a ComK finds itself next to another ComK agent, listed as *neighbor* here:

if *neighbor*=ComK **then**
    random = generate random number between 0 and 1.
    if random < ρ **then**
        neighbor now moves with ComK
    end if
end if

If the probability threshold is not met, then the rule is not executed. The process is repeated at the next time step. Eventually, the probability threshold will be met and the rule is executed. This may take several time steps and is intended to simulate the time it takes for a particular process to occur, i.e., dimerization, transcription, translation, etc.

5.5.3 Parameter Estimation

There are three types of parameters that need to be estimated in this ABM. They are grid environment size, initial concentration of agents, and rule interaction probabilities. In general, the parameters were estimated using a random sweep parameter estimation technique, where parameter estimates were randomly determined followed by repeated simulation runs to validate fit to known experimental results [18].

Rule probabilities were initially estimated based on known interactions and then fit as simulations were run, see Supplementary Information for rule probabilities. For instance, if two molecules have a high affinity then a high

probability of 0.8 was estimated. For two molecules with a low affinity, a low probability of interaction (e.g. 0.2) was initially estimated. As simulations were run, rule probabilities were adjusted to speed up or slow down a particular molecular reaction.

5.5.4 Cell Agent-Based Model

ComK protein production is regulated at transcription by repressor proteins and post-translationally by a protease. The following provides an overview of the molecules involved in the regulation of ComK and is the basis for the Cell ABM.

ComK Transcription: ComK binds at its own promoter as a tetramer acting as its own transcription factor [77, 100]. Random transcription of ComK is a key factor in the build up of large amounts of ComK to trigger transcription of the DNA transport (competence) genes [18, 93]. DegU binds to the *comK* promoter and strongly stimulates binding of ComK dimers to the *comK* promoter [101, 102]. More specifically, DegU binds in between the two ComK dimer binding sites and may possibly facilitate tetramerization of ComK on the *comK* promoter site by partial unwinding and bending of the DNA helix [102]. Transcription can also occur in the absence of ComK [103].

The *comK* promoter site can have several different types of proteins bind to it. This is reflected in the binding rule probabilities, Table 5-1. As DegU promotes ComK dimer binding, the lowest binding probability is used if the promoter agent has no agents bound to it. A higher probability facilitates DegU binding. Finally, the highest probability is used for binding the second ComK dimer if DegU and another ComK dimer are present.

Table 5-1. Interaction probabilities when an agent encounters another agent for the Bind rule.

Agent	Repressor	Promoter	mRNA	ComK	ComS	MecA
Repressor		0.5				
Riobosome			0.9			
DegU		0.5				
ComX		0.5				
ComK		0.5 0.8(+DegU)		0.8		
MecA				0.6	0.7	
ClpP/ClpC						0.5

Transcription probabilities follow a similar fashion (Table 5-2). Transcription will occur at a very low probability when no ComK is bound, increasing to higher values with the addition of one or two bound ComK molecules as shown in Table 5-2. Activators and repressors will disassociate upon successful completion of transcription. After the transcription rule is executed, an mRNA agent is generated, and will persist until it randomly degrades as defined by its death rule. Since the *comK* transcript has a strong Shine-Delgarno ribosome initiation sequence, the binding and translation probability of the ribosome is very high if it encounters a *comK* transcript (Table 5-5). Therefore, the presence of a single transcript is often enough to lead to the production of several ComK proteins.

Table 5-2. Transcription probabilities for *comK* and *comS* promoter agents

Agent	Has Bound	Probability
comK Promoter	ComK tetramer	0.5
	ComK dimer	0.001
	-	0.0001
comS Promoter	ComX	0.5
	-	0.0001

ComK Transcription Regulation: At present there are three known *comK* transcriptional repressors. They are Rok, AbrB and CodY. The *comK* promoter site allows simultaneous binding of AbrB and ComK [104]. The presence of AbrB acts to prevent binding of RNA polymerase as does CodY [105].

In the model, all three repressors are represented by a generic repressor agent that binds to the *comK* promoter agent (Table 5-1). Nutrient limiting conditions down regulate both AbrB and CodY. Thus, in the model, repressor agents are suppressed when nutrients are not available (see below).

ComS Transcription and Regulation: ComS is a protein produced in response to quorum sensing (cell density) [106, 107]. Transcription of *comS* occurs in response to the quorum sensing signaling pathway initiated by the ComX peptide. ComX is produced by the cell at a constant rate during growth and accumulates in the cell medium reflecting cell density [17, 96]. For the purposes of the Cell ABM model the ComX agent is the posttranslationally modified and cleaved extracellular end product, which has been absorbed by the cell. ComX initiates the activation of several proteins, which in turn initiate transcription of *comS* [17]. In conjunction with the interaction probability, the ComX agent represents these events when it binds to the *comS* promoter although in reality it is not the actual *comS* transcription factor.

Implementation of extracellular ComX production is described further below in the Culture Model section.

In addition, regulation of the quorum-sensing pathway is modeled by assuming a Repressor agent acts at the *comS* promoter site, Table 5-1.

ComK and ComS post-translational regulation: The MecA/ClpC/ClpP protease complex degrades both ComK and ComS proteins. MecA, an adapter protein, binds with either ComK or ComS, targeting the proteins for degradation by ClpC/ClpP [94]. ComS competes with ComK for binding with MecA, with ComS having a higher affinity then ComK [95]. If ComK is bound to MecA upon encountering ComS, ComK disassociates, targeting ComS for degradation instead. Because ComK is positively auto-regulated, protection from degradation by ComS results in an explosive increase in ComK synthesis [94]. In this way, the up-regulation of ComS due to quorum sensing leads to an increased accumulation of ComK and transitions to the competence state.

This system was represented by implementing the interaction probabilities shown in Table 5-1. Since ComS has a stronger affinity to the adapter protein MecA, a higher binding probability is used than for association with ComK during the binding rule.

Cell ABM Agents. Agents are translated from the biological model described above to represent ComK, ComS, DegU and MecA proteins, the ComX peptide, ribosomes, ComK and ComS transcripts, repressors, ComK and ComS promoter sites and the ClpC/ClpP protease as shown in Table 5-3. At model startup, a random number of ComK, ComS, ComX and mRNA agents are created. The

number of agents is determined from a random draw within a defined threshold as shown in Table 5-4. The other agent populations are set at a fixed size as listed in Table 5-4. The agents interact with one another and their environment in a 3-D grid of size 40x40x40 cells. Each agent's behavior is defined by a set of rules, summarized in Table 5-3 and described below.

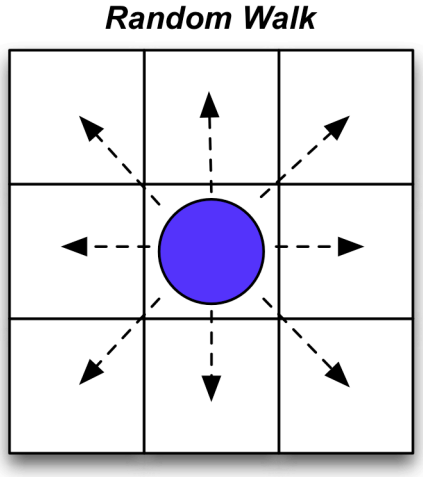

Figure 5.7. 2-D random walk. An agent is moved to a randomly selected adjacent neighboring grid position.

Move Rule. The *move* rule simulates a random walk of an agent throughout the 3-D landscape to simulate molecular diffusion and Brownian motion. At each step, the rule execution results in a one-step move to a randomly selected neighboring cell on the grid, as illustrated in Figure 5.7 for the 2-D case. In a 3-D grid, there are 26 possible adjacent neighboring cells to choose from. The agent will remain in place if the randomly selected cell is occupied.

Table 5-3. Agents and Rules of the Cell ABM

Agent	***Rules***	***Interacts with***
Promoter	transcription	ComK, ComX, DegU, Repressor
Repressor	move, bind	Promoter
Ribosome	move, bind, translation	mRNA
mRNA	move, death	Ribosome
ComK	move, bind	Promoter, ComK, MecA
ComS	move	MecA
ComX	move, bind	Promoter
DegU	move, bind	Promoter
MecA	move, bind,	ComK, ComS, ClpC/ClpP
ClpC/ClpP	move, bind, death	MecA

Table 5-4. Initial concentration of Agents

Agent	**Initial Concentration**
Repressor	12
Promoter	2
mRNA	0
Ribosome	80
DegU	12
ComX	0-12
ComK	0-12
ComS	0-12
MecA	20
ClpP/ClpC	20
Cell	20

Bind rule. This rule is used for molecules that bind with other molecules upon an encounter during the course of the random walk. For each agent,

the *bind* rule searches for other agents at adjacent grid positions. When there is an adjacent agent for which a binding rule is defined, the two can bind and then move together. Exceptions occur when a MecA agent encounters ComS and is already bound to ComK, the latter will disassociate in favor of ComS. Agents who bind with one another are depicted in Table 5-3. Bind rule probabilities are shown in Table 5-1.

Transcription rule. The *Transcription* rule is executed by the Promoter agent and results in the production of mRNA agents. The success of this rule depends on what is currently bound to the Promoter agent, as specified in Table 5-1 and Table 5-2.

Translation rule. The Ribosome agent executes the *Translation* rule generating ComK agents or ComS agents depending on the type of the mRNA agent (Table 5-5).

Table 5-5. Additional rule probabilities

Agent	Death	Translation	Random disassociation
mRNA	0.0001		
ClpC/ClpP	0.5		
Ribosome		0.5	
Repressor			0.0001
DegU			0.0001

Death rule. Both the mRNA agent and the ClpP/ClpC agents implement a *Death* rule. For the mRNA agent, the *Death* rule represents the random degradation of mRNA that occurs in the cell. In the case of ClpP/ClpC protease complex, the *Death* rule initiates the removal/death of the bound ComK or ComS agent. Table 5-5 lists the rule probabilities.

5.5.5 Culture Agent-Based Model

Culture ABM Agents. There is technically only one agent type in the Culture ABM model, a Cell ABM whose internal workings were described above. In the culture model, the Cell ABM acts as an agent with the rules specified in Table 5-6. It interacts with its environment by consuming nutrients, and it interacts with other Cell agents through production and consumption of the ComX pheromone. Nutrients and the ComX peptide are modeled by diffusion equations due to their high concentration. Due to the way Repast Simphony is structured, the Culture plate itself acts as a set of immobile agents to allow for executing the diffusion rule (Table 5-6). Consumption of either nutrients or ComX peptides by the Cell agent are fed to the internal Cell ABMs leading to reduction of Repressor agents or increasing ComX agents, respectively. When a Cell ABM model reaches competence, tracked by the number of ComK agents generated, it is shown as a change in color as shown in Figure 5.4.

Table 5-6. Agents and Rules of the Culture ABM.

Agent	***Rules***	***Interacts with***
Culture Model	diffuse	Nutrients, ComX Peptides
Cell ABM	move, generatePeptide, consumePeptide, consumeNutrients, life, death	Nutrients, ComX Peptides

Cell growth equation. The cell agent implements a growth function to control cell growth, division and death. The growth function is based upon the Logistic Map function: $m_{n+1} = \mu m_n - \mu m^2_n / k$, where $\mu = 0.0058$ is the growth

rate[108], m represents the energy of the cell and k is the maximum energy [109]. m_n is the value of the energy function at iteration n. In the Culture ABM, m_0 was initialized to 5 and k was set to 16. The μm_n term of the equation signifies an energy gain, which occurs when consuming nutrients and the $\mu m^2_n/k$ term decreases the energy as it is assumed that basic metabolism within the cell consumes energy. The calculation of the growth function is split across two rules that are described further below: move and consumeNutrients. The life rule uses the value of the equation to determine whether the cell should divide or not. In the move rule, energy is reduced by $\mu m^2_n/k$. Energy is increased by μm_n upon consumption of a nutrient. If there are no nutrients at the agent's current location, there is no energy increase.

Once nutrients reach a level (<1) such that they can no longer be consumed by Cell agents, the energy level steadily decreases instead of increasing via the move rule. At this stage the cell growth equation is altered and energy is reduced by $d/(k/2)$, where d is the death rate, $d=0.002$.

Culture ABM Rules. In the Culture ABM, agents interact with one another and their environment in a 2-D grid of size 40x40 cells. Rules are executed in a random order for each iteration of the model, and probabilities determine whether or not a rule executes as shown in Table 5-7. Agent behavior is summarized in Table 5-6.

Table 5-7. Cell Agent rules within Culture ABM.

Agent	Death	Move	generatePeptide	consumePeptide	Life
Cell	0.0001	0.5	0.8	0.8	0.8

Diffusion rule. The Culture ABM executes one rule, the *diffusion* rule which is essentially a global rule for the Culture model. This rule executes the Repast Simphony diffusion algorithm on both the nutrient and peptide value layers, based on Rucker's diffusion equation for Cellular Automata [110]. For each cell in the grid the difference between the current cell value and the weighted average of neighboring cells is calculated, multiplied by the diffusion constant and then added to the current cell value, thus, ensuring that the concentrations within the grid cells move down the concentration gradient. The diffusion constant used in this model is 0.1.

Move rule. The Cell agent executes a move rule that simulates bacterial chemotaxis so that the Cell agent moves towards the most favorable nutrient conditions. If there are nutrients available, the Cell agent will remain at its current location. If not, the Cell agent selects a free neighboring location with a higher concentration of nutrients, following the highest nutrient concentration gradient.

The energy reduction portion of the cell growth equation is implemented when this rule is executed. It is assumed that there is a metabolic cost to a cell's energy at each iteration.

The move rule also determines when nutrients can no longer be consumed at the Cell agent's current location and thus disables Repressor agents within the Cell ABM. A randomly selected Repressor agent is then removed from the model if the probability threshold (0.001) is met.

Consumption rules. The consumePeptide and consumeNutrients rules cause the consumption of one molecule from the current grid location of the Cell agent. When a ComX peptide is consumed it is added as an agent to the Cell ABM.

When a nutrient is consumed a gain in energy occurs as described above. The consumeNutrient rule is executed every iteration like all the other rules. However, the consumePeptide rule is executed every 50 iterations if a probability of 0.8 is met.

Generate Peptide Rule. The generatePeptide rule produces one molecule of the ComX peptide. It is added to the concentration at the current grid location. This rule is an approximation of the constant production of the ComX peptide described in the literature [17]. Unlike all other rules, the generatePeptide rule is executed every 100 iterations and a ComX peptide is generated if a probability of 0.8 is met. The ComX peptides produced in this manner are independent of the ComX agents residing within the Cell ABM agent.

Life Rule. The life rule determines whether the cell is ready to divide when the energy exceeds a predefined threshold of 15, which is one less then *k,* the maximum energy (see growth equation parameters above). Cells which exhibit the competence state do not divide. A new Cell ABM is created and added to the grid in a neighboring, adjacent grid location. The daughter cells receive half the energy of the parent cell, and the parent's energy is reduced by half.

To model inheritance, an arbitrary plane is randomly chosen which bisects the parent Cell ABM through its center. Agents 'above' the plane will remain in the parent cell and agents 'below' the plane will go with the daughter cell. The daughter cell is then placed in a randomly selected location adjacent to the parent cell. If that location is occupied then the shove rule is executed.

The *life* rule is also responsible for determining the death of a Cell agent. If the Cell agent's energy is very low (< 0.5), then the Cell agent 'dies' and is removed from the model.

Shove rule. The *shove* rule is intended to displace Cell agents one step to an adjacent, randomly selected, neighboring position if more then one Cell agent occupies its current location. As each agent executes this rule, a one step displacement of Cell agents will ripple through a group of adjacent Cell agents until there is room for all Cell agents on the Culture Model grid.

Death rule. This rule models random die off of cells with a high metabolism when nutrients are insufficient. The death rule is executed every 50 iterations instead of every iteration as in the other rules to decrease the death rate. The Cell agent will 'die' when a probability threshold of 0.0001 is met and when the Cell's energy is within 0.5-7.5 (half the energy threshold needed for division).

Chapter 6
Stochastic Model of BK Virus Replication and Assembly

6.1 Abstract

BK Virus (BKV), a polyomavirus virus in the same family as SV40 and JC Virus, has recently been associated with the Salivary Gland Diseases Sjögrens Syndrome and an HIV associated Salivary Gland Disease. BKV is more infamous for causing the rejection of kidney transplants. As such, BKV infection of salivary gland cells implicates oral transmission of the virus. Thus, a novel, intracellular, computational model using agent-based modeling was developed to model the affects the virus has on the salivary gland cell during BKV's process of replication. In addition to viral proteins, host cell machinery that aids transcription, translation and replication of the BKV genome are modeled. A novel application of the Boids algorithm was implemented to simulate molecular binding and formation of BK virions and BK virus like particles (VLPs). BKV replicates slowly in salivary gland cells, producing infectious virus after 72-96 hours. This model enforces obtained experimental results indicating the processes that result in the slow accumulation of viral proteins. As a result, BKV particles only form after large concentrations of capsid subunits have accumulated.

6.2 Introduction

It is well established that certain viruses can transform a cell and induce the formation of tumors [4]. One such virus is the BK Virus. BKV, a polyomavirus family member, is a non-enveloped, small, double-stranded DNA virus. BKV is believed to cause a harmless latent infection in healthy people but may reactivate if the immune system has been compromised [19]. BKV is known to cause BKV nephropathy (BKN) a kidney transplant complication where reactivated BKV induces cell necrosis due to immunosuppressive drug regimens [111]. BKV sequences have been found in many organs in the human body—kidneys, liver, stomach, lungs, parathyroid glands, lymph nodes, tonsils, lymphocytes, bladder, prostate, uterine cervix, vulva, lips and tongue [112]. Recently, BKV has been detected in HIV positive patients with HIV associated salivary gland disease (HIV SGD) and shown capable of reproducing in salivary gland cells [20]. Since salivary gland diseases such as HIV SGD or Sjögren's Syndrome do not have a known etiological agent, we are pursuing this relationship with BKV using a computational model to reproduce the replication and assembly of BKV within a salivary gland cell.

Complex structures, patterns and phenotypes in biology are often the result of biochemical interactions between molecules. The T=7 icosahedral structure of BKV and other polyomaviruses is an intriguing example of this. Through the interaction of host cell transcription and translation machinery with the BKV genome and the simple interactions of the capsid proteins, a complex icosahedral-shaped BK virion eventually forms. We wish to study the emergent properties of these simple interactions using agent-based modeling (ABM) to determine salivary gland cell processes that influence or hinder virus replication.

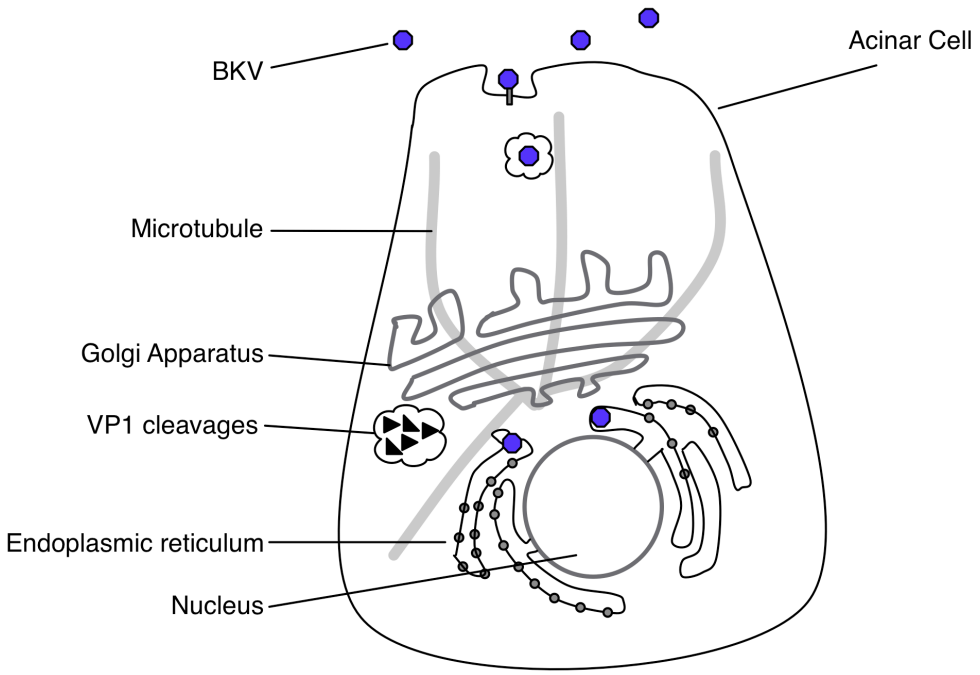

Figure 6.1. Mock-up of BKV entry into a salivary gland cell.

Not a great deal is known about the BKV replication cycle. Much of what is understood about polyomavirus replication has been inferred from studies of the related polyomavirus SV40 [4]. What is understood regarding BKV has come from studies of viral interactions with kidney cells. More recently it has been determined by our group that salivary gland cells are permissive for BKV replication [20]. BKV is believed to enter the cell through caveolae-mediated endocytosis [113, 114] after binding with ganglioside GD1b or GT1b [115, 116] on the cell surface. It is then believed to use the cell's cytoskeleton [117, 118] where it is transported to the ER or Golgi, eventually gathering in the perinuclear region [119]. Recently, it was found that BKV may enter an acidic compartment after entry and travel along microtubules to the ER [120]. BKV disassembly occurs due to VP1 cleavages prior to reaching the ER [120]. BKV egress may occur by cell lysis but BK virions have also been observed

in vesicles in the cytoplasm [119]. The agnoprotein is believed to play a role in nuclear egress [121, 122]. See Figure 6.1 for a representation of BKV entry into an acinar cell.

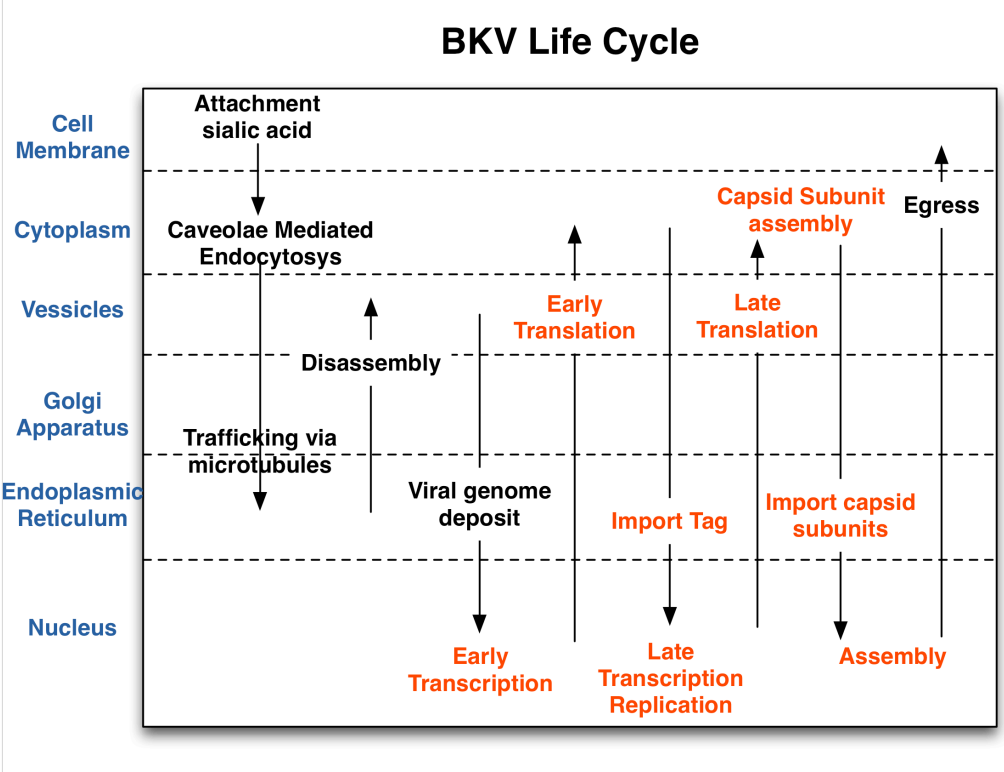

Figure 6.2. Biological and computational model of the BKV Life Cycle reading from left to right. The computational model is indicated in red. The arrows indicate the progress from one compartment to the next.

More is known about attachment and entry of the virus then is known about the other phases of the BKV life cycle. It is not clear (*i*) how the virus releases its genome, (*ii*) how the genome is transported to the nucleus, (*iii*) how BKV assembly occurs in the cell, and (*iv*) how BKV exits the cell. With this initial, single cell model we present the first intracellular model of BKV replication and assembly within a salivary gland cell to address (*iii)*, BKV assembly. Viral transcription and replication using host cell machinery is modeled leading to the eventual assembly of virus like

particles as well as BKV virions via an agent-based model (ABM) to capture viral rates of production as shown in Figure 6.2.

Much of the previous work modeling aspects of viral replication and pathogenesis have used ordinary differential equation (ODE) mathematic models [123-125]. These models have focused on concentrations of the organism within various cellular and extracellular compartments. These models have not considered intracellular function at the level of the organelle or viral gene product. In the study described here, ABM has been used in order to elucidate the aspects of viral transcription, translation and replication.

ABM is a computational modeling technique consisting of autonomous agents interacting with each other and their environment based on a set of simple rules. Each agent is an independent, information carrying, decision-making entity. Biological systems can easily translate to an agent-based model such that proteins, RNAs and other molecules become agents. Agents move by mimicking the erratic random movement of biological elements as if bombarded by molecules in a virtual cellular environment—Brownian motion. Protein interaction networks and metabolic pathways can then be defined by agent rules and agent interaction probabilities. This type of model is inherently stochastic and easily models spatial, temporal interactions between agents based on diffusion/movement rules. In addition, discrete agents are modeled as opposed to a continuous population as in mathematical models of ODEs. ABMs are built from the bottom-up by specifying local individual components such that complex behaviors emerge at the global level. This facilitates the development of multi-scaled models such that an ABM can act as an agent in

another model, e.g., an ABM of a cell incorporated into an ABM consisting of many cells.

For the past two decades, ABMs have been applied to problems in biology mainly within the field of ecology [83]. Here it is often referred to as either individual-based modeling or pattern-oriented modeling [83, 84]. Recently, ABMs have been applied to host pathogen modeling. Duca et al [126] created an ABM to produce a virtual model of the tonsils of the nasopharyngeal cavity and peripheral circulation. The host immune response in granuloma formation in response to tuberculosis infection has also been modeled using an ABM [127]. In these models, agents represent host cells, viral or bacterial cells and cells of the immune system.

While the previous work focuses on cellular concentrations when modeling viral pathogenesis, our model represents the intracellular processes involved in the replication of BKV within a salivary gland cell. In order to model viral replication, a much more detailed agent-based model at the intracellular, molecular level was needed to model molecular interactions within a cell. Host cells essentially determine the growth rate of a virus, and this model mimics the salivary gland cell support and hindrance of the BKV life cycle.

6.3 Results

Using the Repast Simphony agent-based modeling platform, a single-cell salivary gland model initially infected with one BK virion was designed [92]. At present, only known cellular molecules that affect viral transcription and translation are represented as agents in the model. Aspects of calcium, pH, temperature and

salt concentrations which affect the BKV life cycle are currently excluded from the model [4, 120].

A virtual cell consisting of a nucleus and cytoplasm/endoplasmic reticulum (CER) was represented. Within these compartments cellular processes that lead to viral transcription and translation are modeled. Initially, the nucleus compartment contains agents representing the BKV genome, host Transcription Factors and host DNA promoter sites and the CER compartment contains agents representing Ribosomes. Agents change locations from one compartment to another depending upon their function. For instance, mRNA agents that are eventually created are exported from the nucleus to the CER for translation.

6.3.1 Intramolecular interaction

Agent movement is performed by simulating Brownian motion (see Materials and Methods for implementation). Thus, agents binding with other agents are caused by chance encounters due to their random movement. Biologically, intermolecular interactions are governed by binding affinity/repulsion, stoichiometry, conformational change and biochemical reactions. Each of the former can be modeled by the molecular binding and interactions of agents simulated by the Boids Algorithm [128] which is summarized here and detailed further in Materials and Methods. Agent-agent binding begins by imposing simple rules of binding similar to biochemical electrostatic properties of attraction and repulsion followed by a momentum calculation, Figure 6.3. These calculations govern the spherical, icosahedral like formation of the capsid as well as other agent-agent binding described in detail below such that bound agents in the model will move together, in

the same direction with the same velocity, while marinating a separation distance between neighboring agents. These simple rules specify the distance the capsid subunits must maintain from the viral genome agent as well as the distance they must maintain between subunits. This separation distance essentially governs the formation of the capsid and is maintained by the attraction and repulsion calculations to maintain the distance. Repulsion: if the agents are too close together, the agent position is adjusted away. Attraction: if the agents are too far apart, the agent position is adjusted towards the neighboring agent. Momentum of 'bound' agents is maintained by calculating the average velocity of all neighboring agents and adjusting the agent's position towards the average velocity. Visually, the combined position adjustments determined by these rules, results in a slight jitter in the movement of the bound agents while maintaining the formed structure roams randomly around the compartment.

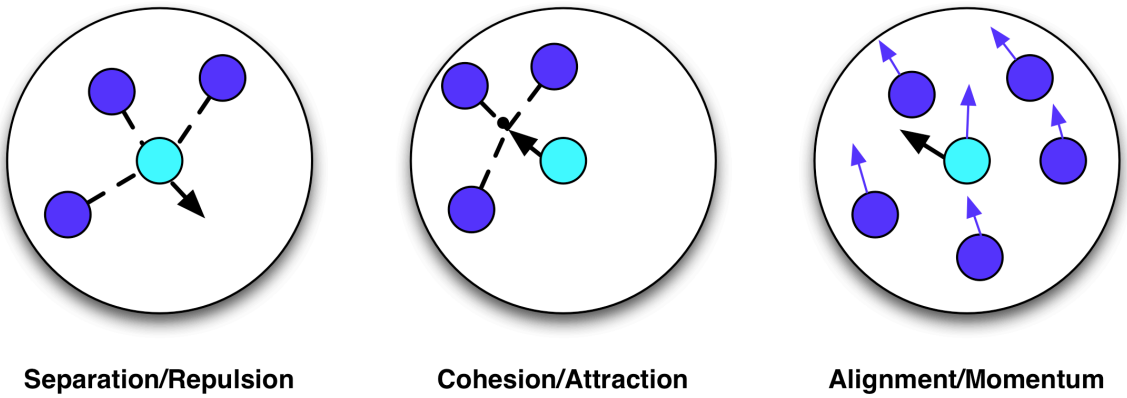

Figure 6.3. Boids rules. Separation/Repulsion: move away from neighbors. Cohesion/Attraction: move towards average position of neighbors. Alignment/Momentum: move towards average heading of neighbors.

6.3.2 Viral protein transcription and translation

Progression of the model is controlled by the regions of the BKV genome as depicted in Figure 6.4. Post entry and uncoating, when a host transcription factor agent binds with the BKV genome agent, early region transcripts are created, Figure 6.4b and Figure 6.5a. Many transcription factors have been identified which bind to the early promoter and they are represented by a single agent type [129]. Alternative splicing is simulated in the model resulting in the production of large T antigen (Tag) transcripts [130]. Small t antigen (tag) transcripts are ignored at present in the model and will be implemented once the cellular pathways tag interacts with have been implemented. For genome replication, Tag binds as a double hexamer and recruits DNA Polymerase to the origin of replication site of the regulatory region [4]. This is simply implemented by the accumulation of Tag agents binding to the BKV genome agent followed by a DNA Polymerase agent binding, resulting in an eventual BKV genome agent, Figure 6.4b and Figure 6.5c. As BKV is a DNA virus, the cell must enter S-phase for DNA polymerase to accumulate initiating BKV genome replication [4]. Tag stimulates entry into S-phase by sequestering pRb resulting in the release of E2F [131, 132]. In the model, DNA polymerase agents are created by the binding of Tag agents to host DNA fragment agents—an indirect representation of this exact mechanism (Fig 5 C). Finally, late region transcription accompanies BKV replication [4]. This is simulated in the model by the accumulation of Tag agents on the BKV genome agent resulting in a late region transcript, Figure 6.4b and Figure 6.5b. Alternative splicing is also simulated resulting in complete VP1, VP2 or VP3 transcripts. Agnoprotein is not produced at this initial phase of the model as it is not required for viral replication but has been implicated in egress [121, 122].

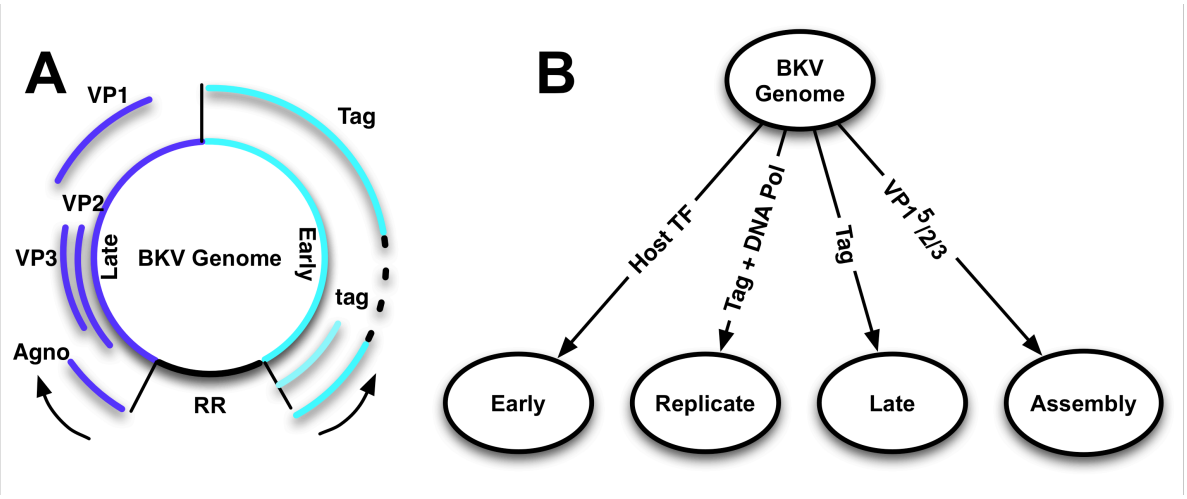

Figure 6.4. A) BKV circular DNA genome depicting the regulatory (RR), early and late regions and transcripts produced. The early region transcribes alternatively splice RNA Tag or tag mRNA. The late region transcribes alternatively splice RNAs for translating to VP1, VP2, VP3 and the agnoprotein. B) The model imitates the transcription of early and late regions based on binding to the regulatory region of host transcription factors, Tag or DNA Polymerase. Capsid assembly begins when a VP1 pentamer is bound.

Viral transcripts are then exported from the nucleus when encountering the boundary between the nucleus and CER compartments. Chance encounters with Ribosome agents with mRNA agents result in the translation of viral protein agents consisting of Tag, VP1, VP2 or VP3, Figure 6.5d. Tag is imported back into the nucleus when the agent encounters the boundary between the compartments. The model assumes that capsid subunits consisting of VP1 pentamers in complex with either VP2 or VP3 are assembled in the CER before being imported into the nucleus as shown in Figure 6.5e [4]. Again, upon encountering the boundary between the two compartments, the capsid subunit complex of a VP1 pentamer and VP2 or VP3 is imported to the nucleus and translated into a single agent for ease of representing the capsid self-assembly process. The import process described is intended to

simulate the nuclear localization signal found on the viral proteins that is recognized for transfer of the protein complex through the nuclear pore complex.

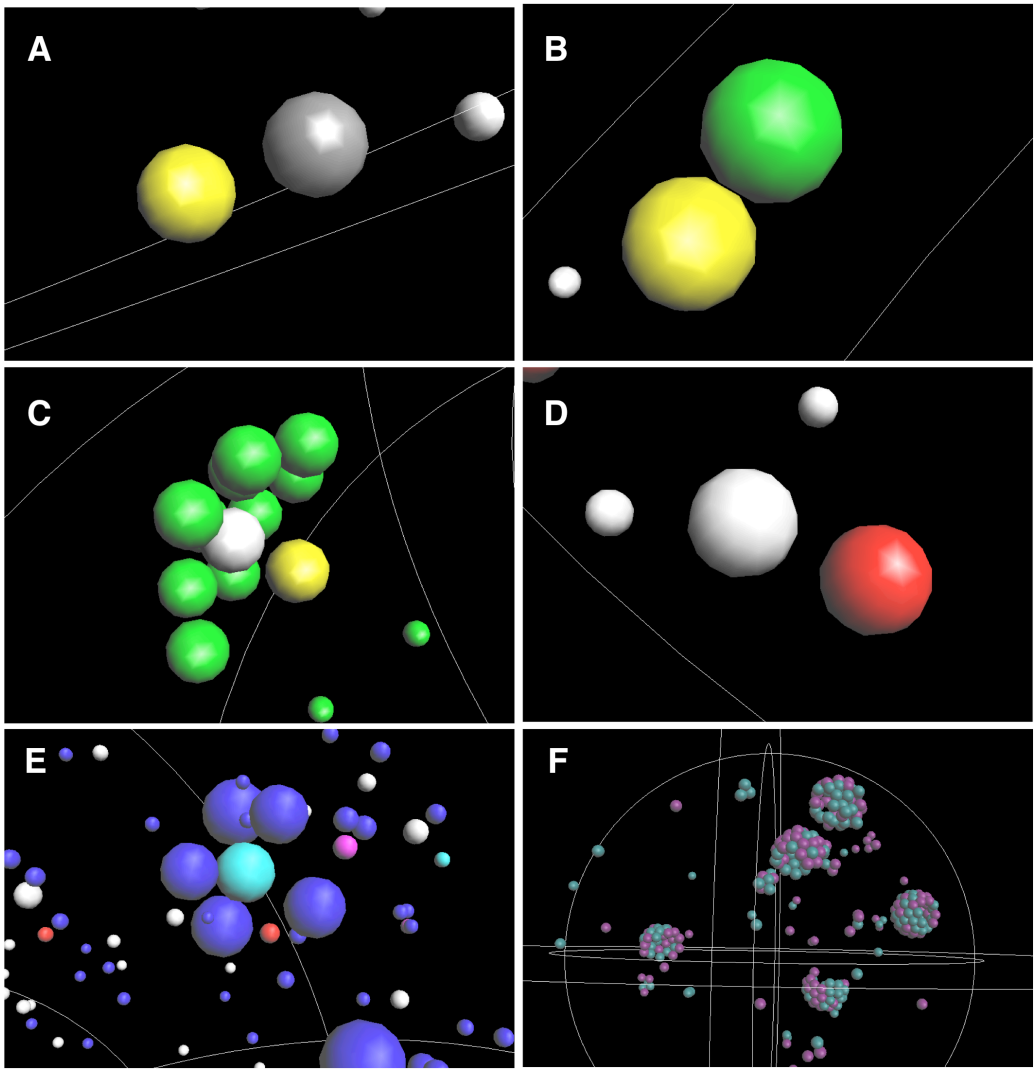

Figure 6.5. Snapshots of agents in a simulation. A) Early transcription: Host Transcription Factor (Grey) binds with BKV Genome (yellow). B) Late transcription: Tag (green) binds with BKV Genome (yellow). C) Genome replication: 12 Tag (green) bind with the BKV genome (yellow) recruit DNA Polymerase (light gray). D) Translation: mRNA (red) binds with Ribosome (white). E) Assembly of capsid subunits. 5 VP1 agents (blue) binding to a VP3 agent (cyan). Agents in the background are ribosomes (white), mRNAs (red) and VP2 (magenta). F) A simulation of only capsid self-assembly showing partial capsid formation of VLPs (4) and virions (1). Purple represents VP2 bound VP1 pentamers and green represents VP3 bound VP1 pentamers. The white lines in the simulation indicate the separation of the nucleus and CER components which are represented spherically.

The model was tuned to results obtained from BKV-salivary gland cell *in vitro* experiments. When monitoring transcript concentrations, a slow ramp-up is observed in the model that coincides with *in vitro* results, Figure 6.6, top row [20].

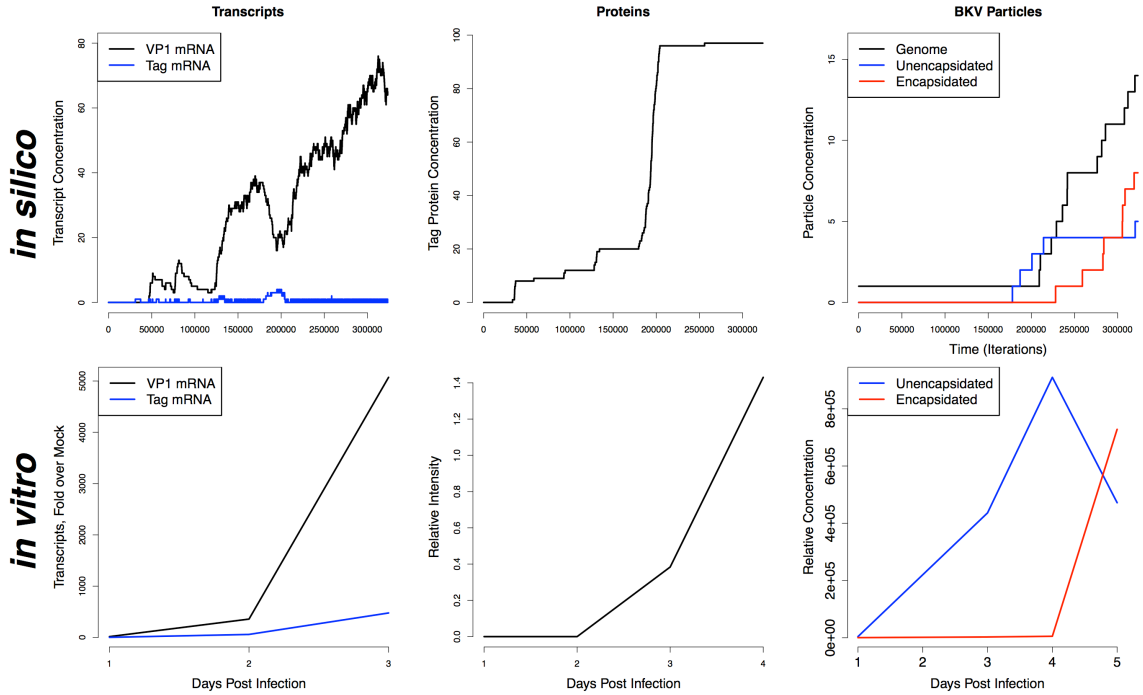

Figure 6.6. *in vitro* and *in silico* results showing transcripts (VP1 and Tag), protein (Tag), genome and BKV particle concentrations in salivary gland cells. *In vitro* western blot (bottom row, middle) of Tag quantified using ImageJ[149].

6.3.3 Virion self-assembly

The last phase of the replication process is encapsidation of the genome. Self-assembly of the BK virion occurs in the nucleus. The capsid is comprised of the structural proteins VP1, VP2 and VP3 and forms a T=7 icosahedron where 12 VP1 pentamers are located in the 12 vertices of the icosahedron surrounded by 5 neighboring VP1 pentamers [133, 134]. 60 pentamers comprise the rest of the structure with 6 neighboring VP1 pentamers [133, 134]. A VP1 pentamer is bound with a VP2 or VP3 protein where the VP2 or VP3 side is presented internally, towards the DNA,

and VP1 is on the external surface of the capsid. The C-terminals of the VP1 proteins extend to bind with neighboring VP1 pentamers solidifying the capsid structure. In this model it is assumed that the $VP1^5$/VP2 or $VP1^5$/VP3 capsomeres are formed in the CER and imported to the nucleus via the VP2 or VP3 nuclear localization signal (NLS). A single agent, VP123, represents the imported capsid subunit.

The model assumes that the capsid subunits can bind DNA and begin assembling around the BKV genome agent relying on genome-subunit and subunit-subunit interactions [135]. However, virus like particles (VLPs) can form in the absence of the viral genome, VP2 and VP3 and thus, aggregation of the subunits leads to the eventual formation of an empty capsid [134]. Empty capsid or VLP formation is modeled by subunit-subunit interactions.

As stated previously, a single agent represents a capsid subunit, a VP1 pentamer bound with either VP2 or VP3. Color coding distinguishes between VP2 or VP3 bound subunits as shown in Figure 6.5b. Assembly begins upon aggregation of capsid subunits and upon encountering the BKV genome agent. Chance encounters of randomly moving capsid agents with the BKV genome agent and with other capsid subunit agents allows the gradual formation of the capsid. Once capsid assembly is completed, the structure will continue to move randomly within the nucleus compartment. The spherical, icosahedral structure formed is enforced by the separation distance defined by subunit-subunit and subunit-genome interaction. Egress is currently simulated when the structure eventually encounters the boundary separating the nucleus from the CER compartments. The VLP or virion is then

removed from the model and a count of particles produced is maintained, Figure 6.6, bottom row.

6.4 Discussion

This work is a novel application of ABM to the modeling of intracellular viral infection and is the first intracellular ABM of BK virus replicating within a salivary gland cell. By modeling molecular interactions within a cell during infection it is possible to understand how the virus affects the function of the cell, to make predictions about therapeutic interventions and to further knowledge about viral pathogenesis in general. This model is a proof of principle that can be applied to the study of many other pathogens and cell types as we have done here with HSG.

Although, modeling with ODEs is a popular technique, it is not as straight forward to translate a biological system into a model as it is for an ABM. In addition, modeling with ODEs assumes high concentrations of molecules, uniform reaction rates and uniform rates of movement all of which affect reaction kinetics [136]. In our ABM, we had at times very small concentrations of agents that could easily be tracked. Also, random spatial and temporal interactions of molecules were easily modeled by our ABM due to the definition of the movement and binding rules simulating real life reaction kinetics. ODEs typically model the average behavior of a system, the ABM models the discrete behavior easily allowing us to scale the model into a system of several individual cell ABMs comprising a tissue and then the organ itself. However, ABMs can be quite computationally intensive, as the number of agents increased in our model, execution speed slowed considerably. We are currently working on improving the computational efficiency of our model.

The definition of simple rules defining agent interaction and binding can model the emergence of complex behavior such as transcription, translation and, most importantly, the self-assembly of viral capsids. The simple interactions between capsid subunits and the viral genome as modeled supports the theory that capsid assembly occurs based on subunit interactions as well as interaction with the viral genome. In this model, interacting with the viral genome enforces the T=7 icosahedral curvature of the capsid. Prior work has shown that without the genome, T=1 structures are possible when reducing disulfide bonds and removing calcium ions [137]. These types of structures are possible with this model by simply modifying the separation distances between subunit agents and genome agents.

The icosahedral structure of BKV and other polyomaviruses is intriguing. Although the protein subunits (VP1 pentamers) have a pentagonal shape, they are packed in 6 neighboring subunits (hexavalent) or 5 neighboring subunits (pentavalent) structures [133, 134, 138, 139]. Several computational models have been built to understand how nature can form such a complex geometrical structure. One such model is built upon the theory that self-assembly of virus capsids is based on the interaction of structural proteins with neighboring protein subunits. Local rules theory creates a model of the virus capsid based on angles and distances between neighboring subunits approximating the conformation changes of the capsid subunits [140-142]. The mathematical problem of pentagon packing has also been applied to the study of the virus structure making distinctions between loosely packed pentagons as in the case of BKV and its polyomavirus cousins and more densely packed pentagons as with papillomaviruses [143].

The ABM presented here shows a more simplified representation of viral self-assembly where capsid subunits maintain a separation distance between neighboring subunits. This results in the emergence of the capsid structure without the need for angle and distance calculations to force the formation of the viral particle. Thus, the formation of the capsid is an emergent property based on the interaction of capsid subunits with neighboring subunits based solely on maintaining a distance from one another. Transcription and translation that also rely on the same simple binding rules resulted in the production of viral proteins.

Further, an additional emergent property of the Boids implementation was a change in the momentum of the forming structure as more and more agents became bound. Structure movement slowed while moving in random directions as would be observed for larger molecules.

The model produced non-infectious particles. In the model, more BK virions were produced then VLPs as shown in Figure 6.6, bottom row, where a maximum of 7 virions and 4 VLP were produced in this example simulation. In addition, it was noticed that as the capsid subunit concentrations increased and began to aggregate, VLPs were formed before the BK genome replicated itself and began to form BK virions. This emphasizes the importance of cooperativity between capsid subuints during particle formation as well as that increased concentrations of capsid subunits encourages particle formation as seen in recent research by Muckherjee et al [144].

This first ABM models the BKV replication process within a single cell. The complete BKV replication cycle will be added to the model. Additional cellular pathways such as the exocrine pathway shall also be added to further the study of

BKV interaction with host cell pathways to theorize methods of viral egress and to also model the break down of this pathway in SGD. Once the single cell is complete, the next phase will incorporate multiple cell ABMs to simulate the infection of the salivary gland. Next, the addition of peripheral circulation and infected lymphocytes to the model will aid in modeling the infection of the human host.

6.5 Materials and Methods

6.5.1 BKV ABM

All the steps in the life cycle of BKV rely on host cell "cooperation". Thus, a representation of a host cell with spherical compartments representing the nucleus within a compartment representing the CER was designed, Figure 6.7. Each compartment contains agents specific to the compartment as well as agents that can traverse between the two, i.e., mRNA agents exported to the CER. The environment of the host cell is represented as a continuous 3-D space such that an agent's location is represented by its x, y and z axis floating point coordinates, i.e., agent *b* is located at location (1.1, 2.05, 30.2).

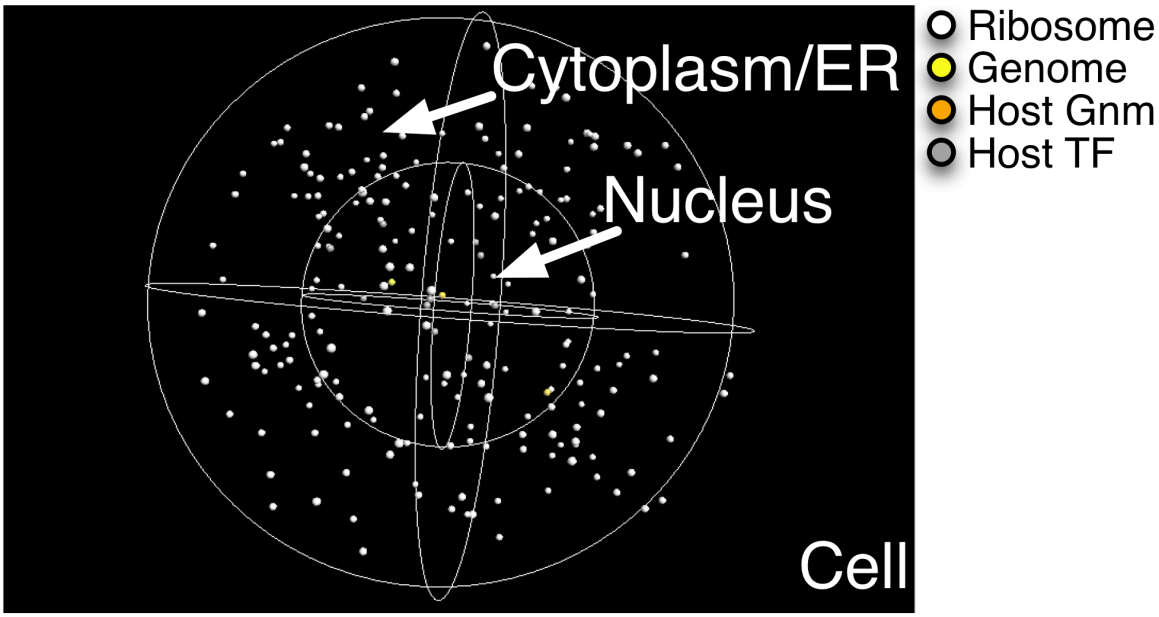

Figure 6.7. Screen capture of model simulation at initial start.

The model is a rather straightforward ABM implemented using Repast Simphony version 1.2 [92] where agents represent various host and viral gene products, Table 6-1. The agents move based on a simulation of Brownian motion in a 3-dimensional environment described below. Once an agent encounters another agent, the movement rule changes to simulate molecular binding by using the Boids algorithm to simulate attraction, repulsion and momentum as detailed further below [128].

Rules are scheduled for execution every clock cycle. However, they are executed in a random order each cycle. In addition, a rule is not executed unless a probability threshold has been met as shown in Table 6-2. As a result it can take several time steps before a rule will successfully execute. Probabilities essentially determine how fast or how slow the production or death of an agent occurs and adjustments were made accordingly.

Table 6-1. The agents and their supported rules.

Agents	*Move*	*Bind*	*Transcription*	*Translation*	*Splice*	*Import*	*Export*	*Death*	*Egress*
V. Genome	x	x	x						x
H. Genome	x	x	x						
DNA Pol	x	x							
Tag	x	x				x			
mRNA	x	x			x		x	x	
Ribosome	x	x		x					
Host TF	x	x							
VP1	x	x							
VP2	x	x				x			
VP3	x	x				x			
VP123	x	x							
VLP	x	x							x

There are three important parameter types that are necessary to estimate in this ABM—size of the environment, initial concentration values of agents and rule probabilities. Parameters were estimated using a random parameter sweep where parameters were randomly determined until the model produced virus and fit *in vitro* data.

At the initial model startup, only Ribosome, Viral Genome, Host Genome and Host Transcription Factor (TF) agents exist, Figure 6.4 and Table 6-2. These agents are placed in random locations within their respective areas, i.e., either the nucleus or the CER. During model execution, newly created agents are placed in a random location within an arbitrarily selected distance of 4 units from the agent creating it.

Table 6-2. Agent rule probabilities and initial concentrations at model startup

Agents	*Initial Concentration*	*Transcription*	*Translation*	*Splice*	*Import*	*Export*	*Death*	*Egress*
V. Genome	1	Early 0.2 Replicate 0.001 Late 0.002						0.5
H. Genome	2	0.001						
H. TF	10							
DNA Pol	0						0.00005	
Ribosome	280		0.4					
mRNA	0			Tag 0.8 tag 0.2 VP1 0.7 VP2 0.2 VP3 0.2		0.4	0.0001	
Tag	0				0.4		0.00005	
VP1	0							
VP2	0				0.9			
VP3	0				0.9			
VP123	0							
VLP	0							0.5

6.5.1.1 Definition of the rules

The viral genome is a circular minichromosome, Figure 6.4a, and can be divided into three regions, which the model is driven by: early, late and regulatory regions (RR). The model follows the early and late transcription biological model by implementing a simple state machine, Figure 6.4b. The output of a state is an mRNA or replicated viral DNA. The input to a state is an agent binding to the regulatory region. Transcription and translation occur by chance encounters of agents that then bind. Successful execution of rules shown in Table 6-1 may result in new agents being created (i.e. proteins in the case of translation) or the changing of compartments.

All agents implement the *move* rule. The move rule is a rather simple simulation of Brownian motion in a 3-dimensional environment where the next position or step the agent makes is determined based on a random draw from a uniform distribution from -0.5 to 0.5 determining the x, y and z coordinates of the direction vector. In all cases, the agents identify the compartment they are to move within and next position calculations take this into account. If a next position calculation oversteps a boundary it is recalculated to reverse the agents direction in order to remain within its compartment. This, in many cases, results in the appearance of the agent "bouncing" off of the boundaries.

Successful execution of the *transcription* rule results in the creation of an mRNA agent placed within an arbitrarily determined distance of 4 units from the genome agent. At this stage the mRNA agent is in an incomplete state and biologically represents an uncapped, unspliced mRNA. Once *transcription* has completed successfully, bound agents will unbind and move freely once again.

The *splice* rule simulates alternative splicing on an incomplete mRNA. The spliced transcript is determined based on probabilities. If a probability is met, the state of the mRNA is marked complete and its type is set. For instance, if the probability for Tag of an early transcript is met, then the mRNA type is set to Tag and marked complete.

The *export* rule is only executed for complete mRNAs and if the mRNA has encountered the boundary between the nucleus and CER containers. Upon successful execution, the agent is moved to the other side of the boundary where it then moves freely within the CER container.

The *translation* rule is executed after the binding of an mRNA agent to a Ribosome agent. Successful execution of the rule results in the creation of a new protein agent depending upon the type of mRNA that is translated. Tag, VP1, VP2 or VP3 agents are created by this rule. The new agent is placed an arbitrarily determined distance of 4 units from the Ribosome agent. The Ribosome agent then unbinds itself from the mRNA agent.

The *import* rule is similar to the *export* rule in that agents are moved from the CER to the nucleus. The rule is executed when the agent encounters the nucleus boundary.

mRNAs are known to randomly degrade, as such a *death* rule was implemented to mimic this functionality. DNA Polymerase and Tag also implement this rule in order to prevent exceedingly high accumulation of the agents to prevent unnecessary consumption of computational resources.

All of the rules described previously are relatively straightforward in the sense that it is executed whether or not a randomly drawn probability from the uniform distribution has met a probability threshold, Table 6-2. However, the *bind* rule is a bit more complex. The bind rule utilizes the Boids algorithm to simulate molecular binding. A typical problem in computer graphics and animation is modeling the movement patterns of flocks of birds and schools of fish in a life-like manner. In 1986, Craig Reynolds developed a simple algorithm called Boids whereby each agent implemented 3 simple position calculations that resulted in a flocking pattern as an emergent behavior [128]. The simple Boids rules were (i) Separation-steering to avoid crowding, (ii) Alignment-steering towards the average heading of neighbors

and (iii) Cohesion-steering to move towards the average position of neighbors, Figure 6.3. Electrostatically speaking these rules can be interpreted as repulsion (separation) and attraction (cohesion) steps in the binding process of two molecules. This algorithm provides a more realistic model of the molecular forces involved in binding.

As stated previously, the *move* rule is essentially a random walk implementation. However, each agent is constantly searching its neighbor space for neighboring agents within a user-defined radius. The *bind* rule is executed in place of the *move* rule when a potential binding partner is found in close proximity. The Boids implementation of the *bind* rule is a simple calculation of the next position of the agent based on position and velocity of neighboring agents. If the agent is too close, the next position is calculated such that it moves away from the other agent a small amount (separation/repulsion). If it is too far away, a position towards the agent is calculated (cohesion/attraction). In addition, the average velocity of the neighboring agents is calculated in order to adjust the velocity of the agent to match its neighbors (alignment/binding). Combined, these rules ensure that the agents move together in the same direction with the same velocity. However, if an agent makes too large of a step when changing position it may disappear from a neighbors view and the agent is then unbound.

VP2, VP3, Viral Genome and VLP agents only implement the alignment/binding phase of the Boids algorithm. This is necessary to facilitate binding of neighboring agents on all sides of the agent.

In sum, the *bind* rule for an agent is a next position calculation based on the distance and velocity of neighboring agents.

VP1 agents in the CER aggregate into pentamers binding with either VP2 or VP3 agents based on successful execution of the *Bind* rule. Once a $VP1^5$/VP2 or $VP1^5$/VP3 complex is successfully formed and its position is close to the nucleus, successful execution of the *import* rule will cause the removal of the 5 VP1 agents and associated VP2 or VP3. The $VP1^5$/VP2 or $VP1^5$/VP3 complex is then represented as a single agent, VP123, to facilitate viral self-assembly within the nucleus.

The curvature of the icosahedral shape of the aggregating capsid subunits is enforced by the distance maintained from the genome agent. To enforce a similar curvature for VLPs, an invisible agent (VLP) is created when 4 or more capsid subunits aggregate. While subunits can bind arbitrarily around the genome agent, subunit-subunit binding is only allowed when forming the VLP around the invisible agent. VP123 agents have a preference for binding with the genome agent over the formation of a VLP.

Lastly, the *egress* rule is essentially a method to identify whether or not a completed virion or VLP has encountered the nucleus boundary. This rule simplifies the as yet unknown process of BKV egress from the cell. The agents involved in the viral complex are then removed from the model—simulating exit from the cell—and a count is kept of virions or VLPs produced.

6.5.2 BKV assays

Material and methods for collection of BKV data assayed in submandibular (HSG), parotid (HSY) and cell lines is as previously described in Jeffers et al [20].

Chapter 7
Conclusion

Modeling complexity does not necessarily require complex techniques. By modeling simple interactions, it is possible to observe complex, emergent behaviors resulting in RNA secondary structure, the competence phenotype in *B. subtilis* or BK viral replication and self-assembly within a host cell.

The simple interactions that form hydrogen bonds between paired nucleotides result in the formation of a complex RNA structure. Determining RNA secondary structure is difficult both experimentally and computationally. The SHAPE data analysis algorithms provided by ShapeFinder provides for a combined chemical and computational system for high-throughput analysis of RNA structure. The algorithms provided by ShapeFinder facilitate the quantification of per nucleotide flexibility and thus, the identification of paired regions of an RNA can be made by structure prediction algorithms such as *RNAstructure* [45, 46]. The statistical analysis strongly affirms that SHAPE does measure nucleotide flexibility and is especially strong in identifying paired regions.

As a result, signal-processing algorithms as implemented in ShapeFinder, described in Chapters 2-4, can be used to aid in the prediction of RNA secondary structure. RNA structure determines its function. As such, the function of a single molecule such as RNAs, enzymes and proteins can then be modeled by using

ABMs. The interactions between molecules leads to the resultant biological pattern, Figure 1.1. ABMs can easily be translated from a biological system by implementing the molecular agents involved in the object of study and defining or predicting how the molecules interact. The simple interactions then result in emergent properties such as the competence phenotype in *B. subtilis* or self-assembly of viral particles. The resultant emergent property is typically shaped by the simple interaction between two or more molecules binding with one another. This interaction between two or more molecules can then result in the repression or activation of a biological function such as transcription, translation and degradation. Simple interactions like molecular binding also result in the self-assembly of large molecular structures as those formed by BKV particles.

ABMs are ideal for modeling molecule-molecule interactions as demonstrated by Chapters 5 and 6. Initially, the ABMs I created for the *B. subtilis* and BKV models were intended for educational purposes in order to understand the system of study by attempting to recreate it. However, even with a well-studied system like competence in *B. subtilis*, there are many aspects that remain unknown that must be hypothesized by the model. This resulted in the identification of additional potential sources of variation in gene expression like spatial-temporal interactions, molecular inheritance and cell division. These were emergent properties from the model I created, which would have been difficult to observe in a top-down model like mathematical modeling.

More complex models evolve from modeling two organisms interacting with one another as in the case of viral infection of a human host as described in Chapter

6. However, these models still rely on simple molecule-molecule interactions. The recreation of the lesser-known system of BKV replication and self-assembly within a host-cell, relied on studies based on BKV and a sister virus, SV40, as well as hypothesized interactions between viral and host molecules, Chapter 6. Much remains unknown about BKV interaction with the human host and the hypotheses implemented by my model were able to make predictions about the production of BK virions and BKV VLPs. My model demonstrated the self-assembly in a 3-D environment of a fully formed viral capsid with or without encapsidating a BKV genome. Again, this was an emergent property that was revealed through simple molecule-molecule interactions that would be difficult to demonstrate in a top-down model such as mathematical modeling. As this model progresses to implement the complete BKV life cycle more hypotheses can be tested in cooperation with *in vitro* experiments eventually leading to the identification of BKV's role in the development of SGD and how it infects the human host.

I used Repast Simphony [92] for the development of the aforementioned ABMs. This ABM platform was selected based on ease of use and 3-D support from a set of ABM platforms that included Swarm [145], Ascape [146] and NetLogo [147]. However, Repast in its current form, is not ideal for the type of biological models I needed to create as Repast is designed for single processor only support. With a model that eventually grew to support approximately 150,000 agents each implementing 3 or more rules, the model execution times exceeded reasonable limits. For example, the *B. subtilis* model can take longer than six weeks to perform approximately 40,000 iterations. Parallel processing is sorely needed and research

in this area will continue in order for model execution to complete within a reasonable time frame.

In addition, techniques to improve the parameter space search in ABMs are not as well developed as they are for mathematical modeling. I used a rudimentary random parameter search technique that was not as efficient as it could have been. Genetic algorithms [148] as well as other techniques used in sociology and ecology ABM would be worth investigating for future models.

References

1. Camazine S. (2001) Self-organization in biological systems. Princeton, N.J.: Princeton University Press. 538 p.

2. Corning PA. (2002) The re-emergence of “emergence”: A venerable concept in search of a theory. Complexity 7(6): 18-30.

3. Alon U, (2007) An introduction to systems biology: Design principles of biological circuits. Boca Raton, FL: Chapman & Hall/CRC, c2007.

4. Fields BN, Knipe DM, Howley PM, Ovid Technologies I. (2007) Fields' virology. Philadelphia: Wolters Kluwer Health/Lippincott Williams & Wilkins. 3091, 86 p.

5. D'Souza V, Summers MF. (2005) How retroviruses select their genomes. Nat Rev Micro 3(8): 643-655.

6. Wilkinson KA, Gorelick RJ, Vasa SM, Guex N, Rein A, et al. (2008) High-throughput SHAPE analysis reveals structures in HIV-1 genomic RNA strongly conserved across distinct biological states. PLoS Biology 6(4): e96 OP.

7. Merino EJ, Wilkinson KA, Coughlan JL, Weeks KM. (2005) RNA structure analysis at single nucleotide resolution by selective 2'-hydroxyl acylation and primer extension (SHAPE). J Am Chem Soc 127(12): 4223-4231.

8. Wilkinson KA, Merino EJ, Weeks KM. (2006) Selective 2[prime]-hydroxyl acylation analyzed by primer extension (SHAPE): Quantitative RNA structure analysis at single nucleotide resolution. Nat Protocols 1(3): 1610-1616.

9. Giddings MC, Severin J, Westphall M, Wu J, Smith LM. (1998) A software system for data analysis in automated DNA sequencing. Genome Res 8(6): 644-665.

10. Vasa SM, Guex N, Wilkinson KA, Weeks KM, Giddings MC. (2008) ShapeFinder: A software system for high-throughput quantitative analysis of nucleic acid reactivity information resolved by capillary electrophoresis. RNA 14(10): 1979-1990.

11. Mortimer SA, Weeks KM. (2007) A fast-acting reagent for accurate analysis of RNA secondary and tertiary structure by SHAPE chemistry. J Am Chem Soc 129(14): 4144-4145.

12. Wilkinson KA, Vasa SM, Deigan KE, Mortimer SA, Giddings MC, et al. (2009) Influence of nucleotide identity on ribose 2′-hydroxyl reactivity in RNA. RNA 15(7): 1314-1321.

13. Efron B. (1979) Bootstrap methods: Another look at the jackknife. The Annals of Statistics 7(1): 1-26.

14. Fisher NI, Hall P. (1990) On bootstrap hypothesis testing. Australian Journal of Statistics 32(22): 177-190.

15. Kaern M, Elston TC, Blake WJ, Collins JJ. (2005) Stochasticity in gene expression: From theories to phenotypes. Nat Rev Genet 6(6): 451-464.

16. Dubnau D, Losick R. (2006) Bistability in bacteria. Mol Microbiol 61(3): 564-572.

17. Draskovic I, Dubnau D. (2005) Competence for genetic transformation. In: Mullany P, editor. The dynamic bacterial genome. Cambridge, UK; New York: Cambridge University Press. pp. 235-273.

18. Maamar H, Raj A, Dubnau D. (2007) Noise in gene expression determines cell fate in bacillus subtilis. Science 317(5837): 526-529.

19. Padgett BL, Walker DL. (1976) New human papovaviruses. Prog Med Virol 22: 1-35.

20. Jeffers LK, Madden V, Webster-Cyriaque J. BK virus has tropism for human salivary gland cells in vitro: Implications for transmission. Virology In Press, Corrected Proof. DOI: 10.1016/j.virol.2009.07.022.

21. Stern S, Moazed D, Noller HF. (1988) Structural analysis of RNA using chemical and enzymatic probing monitored by primer extension. Methods Enzymol 164: 481-489.

22. Ehresmann C, Baudin F, Mougel M, Romby P, Ebel J, et al. (1987) Probing the structure of RNAs in solution. Nucl Acids Res 15(22): 9109-9128.

23. Strobel SA. (1999) A chemogenetic approach to RNA function/structure analysis. Curr Opin Struct Biol 9(3): 346-352. DOI: 10.1016/S0959-440X(99)80046-3.

24. Brenowitz M, R. Chance M, Dhavan G, Takamoto K. (2002) Probing the structural dynamics of nucleic acids by quantitative time-resolved and equilibrium hydroxyl radical 'footprinting'. Curr Opin Struct Biol 12(5): 648-653.

25. Tullius TD, Greenbaum JA. (2005) Mapping nucleic acid structure by hydroxyl radical cleavage. Curr Opin Chem Biol 9(2): 127-134.

26. Maxam AM, Gilbert W. (1980) Sequencing end-labeled DNA with base-specific chemical cleavages. Methods Enzymol 65(1): 499-560.

27. Wilkinson KA, Merino EJ, Weeks KM. (2005) RNA SHAPE chemistry reveals nonhierarchical interactions dominate equilibrium structural transitions in tRNAAsp transcripts. J Am Chem Soc 127(13): 4659-4667.

28. Badorrek CS, Weeks KM. (2005) RNA flexibility in the dimerization domain of a gamma retrovirus. Nat Chem Biol 1(2): 104-111.

29. Badorrek CS, Gherghe CM, Weeks KM. (2006) Structure of an RNA switch that enforces stringent retroviral genomic RNA dimerization. Proceedings of the National Academy of Sciences 103(37): 13640-13645.

30. Badorrek CS, Weeks KM. (2006) Architecture of a gamma retroviral genomic RNA dimer. Biochemistry 45(42): 12664-12672.

31. Gherghe C, Weeks KM. (2006) The SL1-SL2 (stem-loop) domain is the primary determinant for stability of the gamma retroviral genomic RNA dimer. Journal of Biological Chemistry 281(49): 37952-37961.

32. Chen Y, Fender J, Legassie JD, Jarstfer MB, Bryan TM, et al. (2006) Structure of stem-loop IV of tetrahymena telomerase RNA. EMBO J 25(13): 3156-3166.

33. Vicens Q, Gooding AR, Laederach A, Cech TR. (2007) Local RNA structural changes induced by crystallization are revealed by SHAPE. RNA 13(4): 536-548.

34. Dann CE, Wakeman CA, Sieling CL, Baker SC, Irnov I, et al. (2007) Structure and mechanism of a metal-sensing regulatory RNA. Cell 130(5): 878-892.

35. Wang B, Wilkinson KA, Weeks KM. (2008) Complex ligand-induced conformational changes in tRNAAsp revealed by single-nucleotide resolution SHAPE chemistry. Biochemistry 47(11): 3454-3461.

36. Jones CN, Wilkinson KA, Hung KT, Weeks KM, Spremulli LL. (2008) Lack of secondary structure characterizes the 5′ ends of mammalian mitochondrial mRNAs. RNA 14(5): 862-871.

37. Stoddard CD, Gilbert SD, Batey RT. (2008) Ligand-dependent folding of the three-way junction in the purine riboswitch. RNA 14(4): 675-684. 10.1261/rna.736908.

38. Duncan CDS, Weeks KM. (2008) SHAPE analysis of long-range interactions reveals extensive and thermodynamically preferred misfolding in a fragile group I intron RNAâ€ . Biochemistry (N Y) 47(33): 8504-8513.

39. Das R, Laederach A, Pearlman SM, Herschlag D, Altman RB. (2005) SAFA: Semi-automated footprinting analysis software for high-throughput quantification

of nucleic acid footprinting experiments. RNA 11(3): 344-354.

40. Chamberlin SI, Weeks KM. (2000) Mapping local nucleotide flexibility by selective acylation of 2'-amine substituted RNA. J Am Chem Soc 122(2): 216-224.

41. Howell DC. (2002) Statistical methods for psychology. Pacific Grove, CA: Duxbury/Thomson Learning. 802 p.

42. Levenberg K. (1944) A method for the solution of certain problems in least squares. Quart Appl Math 2: 164-168.

43. Marquardt DW. (1963) An algorithm for least-squares estimation of nonlinear parameters. SIAM J Appl Math 11(2): 431-441.

44. R Development Core Team. (2007) R: A language and environment for statistical computing. http://www.r-project.org.

45. Deigan KE, Li TW, Mathews DH, Weeks KM. (2009) Accurate SHAPE-directed RNA structure determination. Proceedings of the National Academy of Sciences 106(1): 97-102.

46. Mathews DH, Disney MD, Childs JL, Schroeder SJ, Zuker M, et al. (2004) Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. PNAS 101(19): 7287-7292.

47. Durbin R, Eddy SR, Krogh A, Mitchison G. (2006) Biological sequence analysis: Probabilistic models of proteins and nucleic acids. Cambridge, U.K.; New York: Cambridge University Press. 368 p.

48. Needleman S, Wunsch C. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 48(3): 443-53.

49. Smith TF, Waterman MS. (1981) Identification of common molecular subsequences. J Mol Biol 147(1): 195-197.

50. Gherghe CM, Shajani Z, Wilkinson KA, Varani G, Weeks KM. (2008) Strong correlation between SHAPE chemistry and the generalized NMR order parameter (S2) in RNA. J Am Chem Soc 130(37): 12244-12245.

51. Jarvinen P, Oivanen M, Lonnberg H. (1991) Interconversion and phosphoester hydrolysis of 2',5'- and 3',5'-dinucleoside monophosphates: Kinetics and mechanisms. J Org Chem 56(18): 5396-5401.

52. Velikyan I, Acharya S, Trifonova A, Foldesi A, Chattopadhyaya J. (2001) The pKa's of 2'-hydroxyl group in nucleosides and nucleotides. J Am Chem Soc

123(12): 2893-2894.

53. Acharya S, Foldesi A, Chattopadhyaya J. (2003) The pKa of the internucleotidic 2'-hydroxyl group in diribonucleoside (3'->5') monophosphates. J Org Chem 68(5): 1906-1910.

54. Li Y, Breaker RR. (1999) Kinetics of RNA degradation by specific base catalysis of transesterification involving the 2â€˜-hydroxyl group. J Am Chem Soc 121(23): 5364-5372.

55. Krasilnikov AS, Yang X, Pan T, Mondragon A. (2003) Crystal structure of the specificity domain of ribonuclease P. Nature 421(6924): 760-764.

56. Tukey JW(W, 1915-2000. Exploratory data analysis. : Reading, Mass.: Addison-Wesley Pub. Co., c1977.

57. Higgins JJ. (2003) An introduction to modern nonparametric statistics. Pacific Grove, CA: Brooks/Cole. 366 p.

58. Westfall PH, Young SS. (1993) Resampling-based multiple testing :Examples and methods for P-value adjustment. New York: Wiley. 340 p.

59. Lawley PD, Shah SA. (1972) Methylation of ribonucleic acids by the carcinogens dimethyl sulphate, N-methyl-N-nitrosourea and N-methyl-N'-nitro-N-nitrosoguanidine. comparisons of chemical analyses at the nucleoside and base levels. Biochem J 128: 117-132.

60. Ehresmann C, Baudin F, Mougel M, Romby P, Ebel J, et al. (1987) Probing the structure of RNAs in solution. Nucl Acids Res 15(22): 9109-9128.

61. Peattie D. (1979) Direct chemical method for sequencing RNA. Proc Natl Acad Sci USA 76(4): 1760-1764.

62. Gilham PT. (1962) An addition reaction specific for uridine and guanosine nucleotides and its application to the modification of ribonuclease action. J Am Chem Soc 84(4): 687-688.

63. Gherghe CM, Mortimer SA, Krahn JM, Thompson NL, Weeks KM. (2008) Slow conformational dynamics at C2'-endo nucleotides in RNA. J Am Chem Soc 130(28): 8884-8885.

64. Lavery R, Pullman A. (1984) A new theoretical index of biochemical reactivity combining steric and electrostatic factors: An application to yeast tRNAPhe. Biophys Chem 19(2): 171-181.

65. Xia T, SantaLucia Jr. J, Burkard ME, Kierzek R, Schroeder SJ, et al. (1998) Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with watson-crick base pairs. Biochemistry (N Y) 37(42): 14719-14735.

66. Lindell M, Romby P, Wagner EGH. (2002) Lead(II) as a probe for investigating RNA structure in vivo. RNA 8(4): 534-541.

67. Soukup GA, Breaker RR. (1999) Relationship between internucleotide linkage geometry and the stability of RNA. RNA 5(10): 1308-1325.

68. David L, Lambert D, Gendron P, Major F. (2001) Leadzyme. Methods Enzymol 341: 518-540. DOI: 10.1016/S0076-6879(01)41174-8.

69. Cannone J, Subramanian S, Schnare M, Collett J, D'Souza L, et al. (2002) The comparative RNA web (CRW) site: An online database of comparative sequence and structure information for ribosomal, intron, and other RNAs. BMC Bioinformatics 3(1): 2.

70. Francis RICC, Manly BFJ. (2001) Bootstrap calibration to improve the reliability of tests to compare sample means and variances. Environmetrics 12(8): 713-729.

71. Henderson IR, Owen P, Nataro JP. (1999) Molecular switches: The ON and OFF of bacterial phase variation. Mol Microbiol 33(5): 919-932.

72. Drenkard E, Ausubel FM. (2002) Pseudomonas biofilm formation and antibiotic resistance are linked to phenotypic variation. Nature 416(6882): 740-743.

73. Smits W, Kuipers O, Veening J. (2006) Phenotypic variation in bacteria: The role of feedback regulation. Nat Rev Micro 4(4): 259-271.

74. Webb JS, Lau M, Kjelleberg S. (2004) Bacteriophage and phenotypic variation in pseudomonas aeruginosa biofilm development. J Bacteriol 186(23): 8066-8073.

75. Maughan H, Nicholson WL. (2004) Stochastic processes influence stationary-phase decisions in bacillus subtilis. J Bacteriol 186(7): 2212-4.

76. Veening J, Smits W, Kuipers O. (2008) Bistability, epigenetics, and bet-hedging in bacteria. Annu Rev Microbiol 62: 193-210.

77. van Sinderen D, Luttinger A, Kong L, Dubnau D, Venema G, et al. (1995) comK encodes the competence transcription factor, the key regulatory protein for competence development in bacillus subtilis. Mol Microbiol 15(3): 455-62.

78. Hamoen LW, Venema G, Kuipers OP. (2003) Controlling competence in bacillus subtilis: Shared use of regulators. Microbiology 149(1): 9-17.

79. Süel G, Garcia-Ojalvo J, Liberman L, Elowitz M. (2006) An excitable gene regulatory circuit induces transient cellular differentiation. Nature 440(7083): 545-550.

80. Süel G, Kulkarni RP, Dworkin J, Garcia-Ojalvo J, Elowitz M. (2007) Tunability and noise dependence in differentiation dynamics. Science 315(5819): 1716-9.

81. Schultz D, Ben Jacob E, Onuchic J, Wolynes P. (2007) Molecular level stochastic model for competence cycles in bacillus subtilis. Proc Natl Acad Sci USA 104(45): 17582-7.

82. Gillespie DT. (1977) Exact stochastic simulation of coupled chemical reactions. J Phys Chem 81(25): 2340-2361.

83. Grimm V, Railsback SF. (2005) Individual-based modeling and ecology. Princeton: Princeton University Press. 428 p.

84. Grimm V. (2005) Pattern-oriented modeling of agent-based complex systems: Lessons from ecology. Science 310(5750): 987-991.

85. Kreft JU, Picioreanu C, Wimpenny JW, van Loosdrecht MC. (2001) Individual-based modelling of biofilms. Microbiology 147: 2897-912.

86. Goryachev A, Toh D, Wee K, Lee T, Zhang H, et al. (2005) Transition to quorum sensing in an agrobacterium population: A stochastic model. PLoS Comput Biol 1(4): e37.

87. Xavier J, Foster K. (2007) Cooperation and conflict in microbial biofilms. Proc Natl Acad Sci USA 104(3): 876-81.

88. Johnson LR. (2008) Microcolony and biofilm formation as a survival strategy for bacteria. J Theor Biol 251(1): 24-34.

89. Nadell C, Xavier J, Levin S, Foster K. (2008) The evolution of quorum sensing in bacterial biofilms. Plos Biol 6(1): e14.

90. D'Agata EM, Magal P, Olivier D, Ruan S, Webb GF. (2007) Modeling antibiotic resistance in hospitals: The impact of minimizing treatment duration. J Theor Biol 249(3): 487-99.

91. Murphy JT, Walshe R, Devocelle M. (2008) A computational model of antibiotic-resistance mechanisms in methicillin-resistant staphylococcus aureus (MRSA). J

Theor Biol 254(2): 284-93.

92. Repast Organization for Architecture and Design. (2008) Recursive porous agent simulation toolkit (repast). http://repast.sourceforge.net/.

93. Leisner M, Stingl K, Radler JO, Maier B. (2007) Basal expression rate of comK sets a 'switching-window' into the K-state of bacillus subtilis. Mol Microbiol 63(6): 1806-1816.

94. Turgay K, Hahn J, Burghoorn J, Dubnau D. (1998) Competence in bacillus subtilis is controlled by regulated proteolysis of a transcription factor. EMBO J 17(22): 6730-8.

95. Prepiak P, Dubnau D. (2007) A peptide signal for adapter protein-mediated degradation by the AAA+ protease ClpCP. Mol Cell 26(5): 639-647.

96. Schneider KB, Palmer TM, Grossman AD. (2002) Characterization of comQ and comX, two genes required for production of ComX pheromone in bacillus subtilis. J Bacteriol 184(2): 410-419.

97. Veening J, Stewart EJ, Berngruber TW, Taddei F, Kuipers OP, et al. (2008) Bet-hedging and epigenetic inheritance in bacterial cell development. Proc Natl Acad Sci USA 105(11): 4393-4398.

98. Levin B, Rozen D. (2006) Non-inherited antibiotic resistance. Nat Rev Micro 4(7): 556-62.

99. Keren I, Kaldalu N, Spoering A, Wang Y, Lewis K. (2004) Persister cells and tolerance to antimicrobials. FEMS Microbiol Lett 230(1): 13-8.

100. Hamoen LW, Van Werkhoven AF, Bijlsma JJE, Dubnau D, Venema G. (1998) The competence transcription factor of bacillus subtilis recognizes short A/T-rich sequences arranged in a unique, flexible pattern along the DNA helix. Genes Dev 12(10): 1539-1550.

101. Ogura M, Tanaka T. (1996) Bacillus subtilis DegU acts as a positive regulator for comK expression. FEBS Lett 397(2-3): 173-176.

102. Hamoen L, Van Werkhoven A, Venema G, Dubnau D. (2000) The pleiotropic response regulator DegU functions as a priming protein in competence development in bacillus subtilis. Proc Natl Acad Sci USA 97(16): 9246-51.

103. Smits W, Hoa T, Hamoen L, Kuipers O, Dubnau D. (2007) Antirepression as a second mechanism of transcriptional activation by a minor groove binding protein. Mol Microbiol 64(2): 368-381.

104. Hamoen LW, Kausche D, Marahiel MA, van Sinderen D, Venema G, et al. (2003) The bacillus subtilis transition state regulator AbrB binds to the −35 promoter region of comK. FEMS Microbiol Lett 218(2): 299-304.

105. Ratnayake-Lecamwasam M, Serror P, Wong K, Sonenshein AL. (2001) Bacillus subtilis CodY represses early-stationary-phase genes by sensing GTP levels. Genes Dev 15(9): 1093-1103.

106. D'Souza C, Nakano MM, Zuber P. (1994) Identification of comS, a gene of the srfA operon that regulates the establishment of genetic competence in bacillus subtilis. Proc Natl Acad Sci U S A 91(20): 9397-9401.

107. Hamoen LW, Eshuis H, Jongbloed J, Venema G, Sinderen D. (1995) A small gene, designated comS, located within the coding region of the fourth amino acid-activation domain of srfA, is required for competence development in bacillus subtilis. Mol Microbiol 15(1): 55-63.

108. Burdett ID, Kirkwood TB, Whalley JB. (1986) Growth kinetics of individual bacillus subtilis cells and correlation with nucleoid extension. J Bacteriol 167(1): 219-230.

109. Fall CP, Marland ES, Wagner JM, Tyson JJ. (2004) Computational cell biology. New York, NY: Springer-Verlag New York, Inc. 468 p.

110. Rucker R. Continuous-valued cellular automata in two dimensions. http://www.cs.sjsu.edu/faculty/rucker/capow/santafe.html.

111. Nickeleit V, Singh HK, Mihatsch MJ. (2003) Polyomavirus nephropathy: Morphology, pathophysiology, and clinical management. Curr Opin Nephrol Hypertens 12(6): 599-605.

112. Tognon M, Corallini A, Martini F, Negrini M, Barbanti-Brodano G. (2003) Oncogenic transformation by BK virus and association with human tumors. Oncogene 22(33): 5192-5200.

113. Eash S, Querbes W, Atwood WJ. (2004) Infection of vero cells by BK virus is dependent on caveolae. J Virol 78(21): 11583-11590.

114. Moriyama T, Marquez JP, Wakatsuki T, Sorokin A. (2007) Caveolar endocytosis is critical for BK virus infection of human renal proximal tubular epithelial cells. J Virol 81(16): 8552-8562.

115. Low JA, Magnuson B, Tsai B, Imperiale MJ. (2006) Identification of gangliosides GD1b and GT1b as receptors for BK virus. J Virol 80(3): 1361-1366.

116. Dugan AS, Eash S, Atwood WJ. (2005) An N-linked glycoprotein with {alpha}(2,3)-linked sialic acid is a receptor for BK virus. J Virol 79(22): 14442-14445.

117. Eash S, Atwood WJ. (2005) Involvement of cytoskeletal components in BK virus infectious entry. J Virol 79(18): 11734-11741.

118. Moriyama T, Sorokin A. (2008) Intracellular trafficking pathway of BK virus in human renal proximal tubular epithelial cells. Virology, 371(2): 336-349.

119. Drachenberg CB, Papadimitriou JC, Wali R, Cubitt CL, Ramos E. (2003) BK polyoma virus allograft nephropathy: Ultrastructural features from viral cell entry to lysis. American Journal of Transplantation 3(11): 1383-1392.

120. Jiang M, Abend JR, Tsai B, Imperiale MJ. (2009) Early events during BK virus entry and disassembly. J Virol 83(3): 1350-8.

121. Okada Y, Suzuki T, Sunden Y, Orba Y, Kose S, et al. (2005) Dissociation of heterochromatin protein 1 from lamin B receptor induced by human polyomavirus agnoprotein: Role in nuclear egress of viral particles. EMBO Rep 6(5): 452-7.

122. Johannessen M, Myhre M, Dragset M, Tummler C, Moens U. (2008) Phosphorylation of human polyomavirus BK agnoprotein at ser-11 is mediated by PKC and has an important regulative function. Virology 379(1): 97-109.

123. Perelson AS, Neumann AU, Markowitz M, Leonard JM, Ho DD. (1996) HIV-1 dynamics in vivo: Virion clearance rate, infected cell life-span, and viral generation time. Science 271(5255): 1582-1586.

124. Ribeiro RM, Lo A, Perelson AS. (2002) Dynamics of hepatitis B virus infection. Microbes and Infection, 4(8): 829-835.

125. Kepler GM, Nguyen HK, Webster-Cyriaque J, Banks HT. (2007) A dynamic model for induced reactivation of latent virus. J Theor Biol 244(3): 451-462.

126. Duca KA, Shapiro M, Delgado-Eckert E, Hadinoto V, Jarrah AS, et al. (2007) A virtual look at epstein-barr virus infection: Biological interpretations. PLoS Pathog 3(10): e137.

127. Segovia-Juarez JL, Ganguli S, Kirschner D. (2004) Identifying control mechanisms of granuloma formation during M. tuberculosis infection using an agent-based model. J Theor Biol 231(3): 357-376.

128. Reynolds CW. (1987) Flocks, herds, and schools: A distributed behavioral model. Comput Graph 21(4): 25-34.

129. Moens U, Vanghelue M. (2005) Polymorphism in the genome of non-passaged human polyomavirus BK: Implications for cell tropism and the pathological role of the virus. Virology 331(2): 209-231.

130. Eash S, Manley K, Gasparovic M, Querbes W, Atwood W. (2006) The human polyomaviruses. Cellular and Molecular Life Sciences 63(7): 865-876.

131. Ludlow JW, DeCaprio JA, Huang C, Lee W, Paucha E, et al. (1989) SV40 large T antigen binds preferentially to an underphosphorylated member of the retinoblastoma susceptibility gene product family. Cell, 56(1): 57-65.

132. Lin JY, Simmons DT. (1991) The ability of large T antigen to complex with p53 is necessary for the increased life span and partial transformation of human cells by simian virus 40. J Virol 65(12): 6447-6453.

133. Liddington R, Yan Y, Moulai J, Sahli R, Benjamin T, et al. (1991) Structure of simian virus 40 at 3.8-A resolution. Nature 354(6351): 278-84.

134. Li T, Takeda N, Kato K, Nilsson J, Xing L, et al. (2003) Characterization of self-assembled virus-like particles of human polyomavirus BK generated by recombinant baculoviruses. Virology 311(1): 115-124.

135. Roitman-Shemer V, Stokrova J, Forstova J, Oppenheim A. (2007) Assemblages of simian virus 40 capsid proteins and viral DNA visualized by electron microscopy. Biochemical and Biophysical Research Communications 353(2): 424-430.

136. Materi W, Wishart DS. (2007) Computational systems biology in drug discovery and development: Methods and applications. Drug Discov Today 12(7-8): 295-303.

137. Nilsson J, Miyazaki N, Xing L, Wu B, Hammar L, et al. (2005) Structure and assembly of a T=1 virus-like particle in BK polyomavirus. Journal of Virology 79(9): 5337-45.

138. Baker T, Drak J, Bina M. (1989) The capsid of small papova viruses contains 72 pentameric capsomeres: Direct evidence from cryo-electron-microscopy of simian virus 40. Biophys J 55(2): 243-53.

139. Griffith J, Griffith D, Rayment I, Murakami W. (1992) Inside polyomavirus at 25-aa resolution. Nature 355:652-4.

140. Berger B, Shor P, Tucker-Kellogg L, King J. (1994) Local rule-based theory of virus shell assembly. Proc Natl Acad Sci USA 91(16): 7732-6.

141. Schwartz R, Shor P, Prevelige P, Berger B. (1998) Local rules simulation of the kinetics of virus capsid self-assembly. Biophys J 75(6): 2626-36.

142. Schwartz R, Garcea RL, Berger B. (2000) "Local rules" theory applied to polyomavirus polymorphic capsid assemblies. Virology 268(2): 461-470.

143. Tarnai T, Gaspar Z, Szalai L. (1995) Pentagon packing models for" all-pentamer" virus structures. Biophysical Journal 69:612-8.

144. Mukherjee S, Abd-El-Latif M, Bronstein M, Ben-nun-Shaul O, Kler S, et al. (2007) High cooperativity of the SV40 major capsid protein VP1 in virus assembly. PLoS ONE 2(1): e765.

145. Swarm Development Group. (1999) Swarm. http://www.swarm.org.

146. Inchiosa ME, Parker MT. (2002) Overcoming design and development challenges in agent-based modeling using ASCAPE. Proceedings of the National Academy of Sciences 99(90003): 7304-7308.

147. Wilensky U. (1999) NetLogo. http://ccl.northwestern.edu.libproxy.lib.unc.edu/netlogo/.

148. Calvez B, Hutzler G. (2005) Parameter space exploration of agent-based models. Knowledge-Based Intelligent Information and Engineering Systems 3684:633-639.

149. Abramoff PJ, Magelhaes PJ, Ram SJ. (2004) Image processing with Image. J. Biophotonics Intl. 11(2004):36-42.