# RNA Optimization in Flow Reactors
## A study *in silico*

DISSERTATION

eingereicht von

**Andreas Wernitznig**

zur Erlangung des akademischen Grades

Doctor rerum naturalium

an der Formal- und Naturwissenschaftlichen Fakultät

der Universität Wien

24. Oktober 2001

Allen die zum Gelingen dieser Doktorarbeit beigetragen haben, möchte ich herzlich danken:

Peter Schuster danke ich für die interessante Themenstellung und Betreuung. Trotz seines vollen Terminkalenders hat er immer Zeit für mich gefunden.

Walter Fontana danke ich für die wichtige Einführung und intensive Betreuung in den ersten Monaten. Auch nachdem er unsere Arbeitsgruppe verlassen hat, hat er mich immer wieder inspiriert und motiviert.

Meine Freunde Günther Weberndorfer, Alexander Renner und Stephan Kopp waren bereit, die Bioinformatik gemeinsam mit mir zu unserem Beruf zu machen.

Viele interessante Diskussionen und ein freundschaftliches Verhältnis verbinden mich ich mit Christoph Flamm, Christian Haslinger, Stefan Müller und Stefan Wuchty.

Ivo Hofacker und Peter Stadler haben immer fachkundige Anworten auf meine Fragen gefunden.

Allen anderen Kollegen, Bärbel Krankhofer, Bärbel Stadler, Thomas Griesmacher, Susanne Rauscher, Andreas Svrcek-Seiler, Jan Cupal, Judith Jakubetz, Christina Witwer, Kurt Grünberger, Jörg Hackermüller, Michael Kosbach, Ulli Mückstein, Roman Stocsits, Caroline Thurner, Stefanie Widder, Michael Wolfinger, Daniela Dorigoni, Norbert Tschulenk, Martin Fekete, Ronke Babajide, Ingrid Abfalter und Dagmar Friede danke ich für die angenehme Arbeitsathmosphäre.

Ganz besonders danken möchte ich meiner Familie: Meine Frau Tanja unterstützte mich in allen Belangen. Sie ist immer eine geduldige Zuhörerin und Partnerin, hilft mir, nicht meine gesamte Zeit vom dem Computer zu verbringen und verhinderte nicht zuletzt den finanziellen Ruin. Meine Tochter Annika, die mich nur als Doktoranden kennt, hat mir gezeigt, daß es wichigere Dinge im Leben gibt.

Ich danke auch meinen Eltern, die mich immer unterstützt haben und mir viele sorglose Jahre beschert haben.

## Abstract

RNA folding gives rise to the simplest known genotype-phenotype mapping. Based on this mapping the simulation of evolutionary processes in flow reactors provides insights into the mechanisms of replication, mutation, and selection in the huge space of possible RNA sequences. Evolution in the flow reactor is relatively easy to implement *in silico*. Based on an algorithm due to Daniel Gillespie a computational flow reactor class library and analysis package was conceived, developed, and implemented during this thesis. Using these programs the evolutionary consequences of replication-mutation dynamics based on RNA sequences have been studied.

In silico evolution in the flow reactor shows pronounced punctuation. Relatively short adaptive phases are interrupted by long epochs of phenotypic stasis. Similar to experimental data the speed of spreading in shape space and sequence space during evolutionary optimization is opposite: Genotypic evolution is faster during quasi-stationary epochs, whereas it slows down in the adaptive phases. The graphical representation using principal components analysis shows the formation of distinct clusters. According to a cluster analysis (*subgraph clustering*) they have no sequence exchange with other clusters.

*Relay series* are a form of backtracking of the relevant shapes from the target structure to a structure of the initial population. With a low mutation rate of $p = 0.001$, rare events of major structural changes (discontinuous transitions) can be observed as well as many continuous transitions in the neighbourhood of the dominating neutral networks. The number of discontinuous transitions in such a relay series stays astonishingly constant with different population sizes while the number of continuous transitions varies. Modeled by means of a linear birth-and-death process larger populations allow an uninterrupted existence of relevant shapes. In small populations, however, they die out and have to be recreated again.

The performance of serial transfer experiments in comparison to continuous flow reactors was examined as well: Especially with a high reduction step and a high maximal population size the serial transfer experiments perform

significantly worse than flow reactors.

At high mutation rates up to the error threshold the share of sequences with maximal fitness is independent of the population size, whereas the number of replications to reach the target shape is higher on average in larger flow reactor populations.

An exact way to explore the succession of RNA sequences in an evolutionary simulation is the *lineage series*. Comparing relay and lineage series on the level of shapes reveals a remarkable difference between the two successions: Both series coincide during discontinuous transitions whereas they follow different paths in the continuous case.

## Zusammenfassung

RNA Faltung ermöglicht die einfachste relevante Genotyp-Phänotyp Abbildung die wir kennen. Basierend auf dieser Abbildung ermöglicht die Simulation von evolutionären Prozessen Einsichten in die Mechanismen von Replikation, Mutation und Selektion im riesigen Raum von möglichen RNA-Sequenzen. Evolution in Flußreaktoren ist relativ leicht *in silico* zu implementieren.

Mit Hilfe eines Algorithmus von Daniel Gillespie wurde während dieser Arbeit eine Flußreaktor-Klassen-Bibliothek und ein Analysepaket entwickelt und implementiert. Mit Hilfe dieser Programme wurden die evolutionären Konsequenzen von Replication-Mutations-Dynamik basierend auf RNA-Sequenzen studiert.

*In silico* Evolution im Flußreaktor hat auffällige Merkmale: Relativ kurze adaptive Phasen werden von langen Abschnitten von phänotypischer Stasis unterbrochen. Ähnlich zu experimentellen Daten ist die Geschwindigkeit der Ausbreitung im Sequenz- bzw. Strukturraum während dieser evolutionären Simulation unterschiedlich: Während phänotypischer Stasis ist die Bewegung von Genotypen erhöht, hingegen sind große Strukturänderungen von relativ kleinen RNA-Sequenzveränderungen begleitet. Die grafische Darstellung mit Hilfe der Prinzipal-Komponenten-Analyse zeigt die Bildung von getrennten Clustern. Eine Cluster-Anaylse (*Subgraph Clusterung*) zeigt, daß kein Sequenzaustausch mit anderen Clustern passiert.

Die *relay series* sind eine Art Rückwanderung der relevanten Strukturen von der Zielstruktur zu einer Struktur der Anfangspopulation. Bei einer niedrigen Mutationsrate von $p = 0.001$ können sowohl seltene, große Strukturänderungen (diskontinuierliche Übergänge) als auch häufige, kontinuierliche Übergänge in der Nachbarschaft der dominierenden neutralen Netze beobachtet werden. Die Anzahl der diskontinuierlichen Übergänge in einer solchen *relay series* bleibt bei unterschiedlicher Populationsgröße erstaunlich konstant, während die Anzahl an kontinuierlichen Übergänge stark variiert. Mithilfe eines linearen Geburts-und-Todes-Prozesses wurde gezeigt, daß größere Populationen die ununterbrochene Existenz von relevanten Strukturen erlau-

ben, während in kleinen Populationen diese immer wieder neu gebildet werden müssen.

Des weiteren wurde die Leistung von Seriell-Transfer Experimenten im Vergleich zu kontinuierlichen Flußreaktoren ermittelt: Die Seriell-Transfer Experimente mit großem Reduktionsschritt und mit großer maximaler Populationsgröße leisten signifikant weniger als Flußreaktoren.

Mit einer hohen Mutationsrate bis zur sog. Fehlerschwelle ist der Anteil an Sequenzen mit maximaler Fitneß unabhängig von der Populationsgröße, hingegen ist die Anzahl an Replikationen um die Zielstruktur zu finden in großen Reaktoren im Durchschnitt höher.

Eine exakte Methode um die Vererbungsabfolge von RNA-Sequenzen in einer Evolutionssimulation zu ermitteln ist die sog. Abstammungsserie (*lineage series*). Wenn man *relay series* und Abstammungsserie auf der Ebene der Strukturen vergleicht gibt es bemerkenswerte Unterschiede: Beide stimmen während diskontinuierlicher Übergänge überein, hingegen folgen sie im kontinuierlichen Fall unterschiedlichen Pfaden.

# Contents

# Nomenclature

| | |
|---|---|
| $\ell$ | sequence lenght |
| $d_{ij}^{h}$ | Hamming distance between RNA sequence strings $i$ and $j$ |
| $d_{ij}^{s}$ | Hamming distance between RNA secondary structure $i$ and $j$ in parenthesis notation |
| $N_{set}$ | given average population size of a flow reactor run |
| in silico | computer memory for a simulation of a natural process |
| in vitro | an artificial environment for natural processes |
| mfe | minimum free energy |
| RNA | ribonucleic acid |
| sequence | RNA sequence type with a distinct order of nucleotids |
| TSP | traveling salesman problem |

# 1 Introduction

In this chapter we describe first in section 1.1 the principles of serial transfer experiments and flow reactors. Then two important experiments of *in vitro* evolution are presented and analyzed. The next section 1.2 deals with a two stage assignment of fitness to genotypes. The evolution of populations, based on chemical kinetics is the topic of the following section. After a short summary about prediction of RNA secondary structure the generic properties of folding are described. We continue by discussing the *relay series* which is the succession of secondary structures generating each other and eventually end by a definition of nearness in shape space.

## 1.1 Serial Transfer Experiments and Flow Reactors

If microbial *in vitro* experiments are carried out in a batch reactor, the incubated cells consume the available nutrients after some time. Due to lack of energy (or production of toxins) the reactor population decelerates metabolism and eventually dies out. The culture may be kept in a growth phase by application of two different strategies. (i) In serial transfer experiments, a subsample of the culture is repeatedly transferred to fresh stock solution. (ii) In a flow reactor fresh stock solution is added continuously and allow an equal volume of culture to drain from the vessel to achieve chemostatic or turbidostatic conditions.

$Q\beta$ is a bacteriophage, whose 4 200 nucleotides code for 4 different proteins. One of these proteins is a subunit of a highly specific replicase with an error rate of $3 \times 10^{-4}$. In the late 1960s Sol Spiegelman and coworkers purified this replicase to accomplish an epoch making experiment, simulating molecular evolution in a serial transfer experiment.

In a stock solution containing RNA replicase and activated nucleotide monomers (ATP, UTP, GTP and CTP) Spiegelman and coworkers incubated

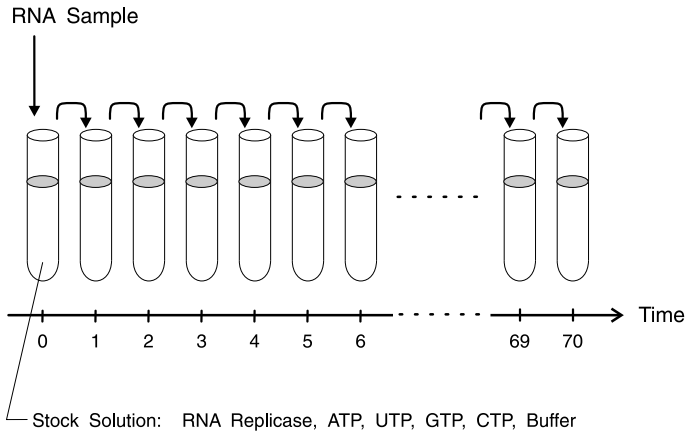



Figure 1.1: Evolution during a serial transfer experiment.

RNA from $Q\beta$, as a template, which is optimized to fulfil its function as an RNA virus genome. Since this reproduction system, in comparison to others, is not very accurate, in the following RNA replication many mutants are created. After the nucleotide monomers are consumed, a small sample of the reaction mixture is transferred to fresh stock solution (dilution 1:12.5). This procedure is repeated some fifty to hundred times. During every serial transfer step the RNA molecules with highest replication rates use up the nucleotides first, while the slower ones in average get little chance to reproduce itself. Therefore the fastest replicators are amplified preferentially in the test tube, become selected with high probability by the next transfer step and inherit their properties to the forthcoming generations. The mutants of the original RNA are inferior replicators and therefore lower the overall replication rate first. This is due to the large scale deletions that occur during the initial phase. But after some time, faster replicators are found increasing RNA synthesis rate by more than one order of magnitude [49]. These RNAs are not infective anymore but they are optimized for fast replication.

Another serial transfer experiment was carried out by Richard Lenski and coworkers [11, 31, 36]. Bacteria populations of $5 \times 10^8$ cells where diluted 1:100 by fresh medium every day for about three years. An average generation time of 2.6 hours results in about 10 000 generations. During the initial 2 000

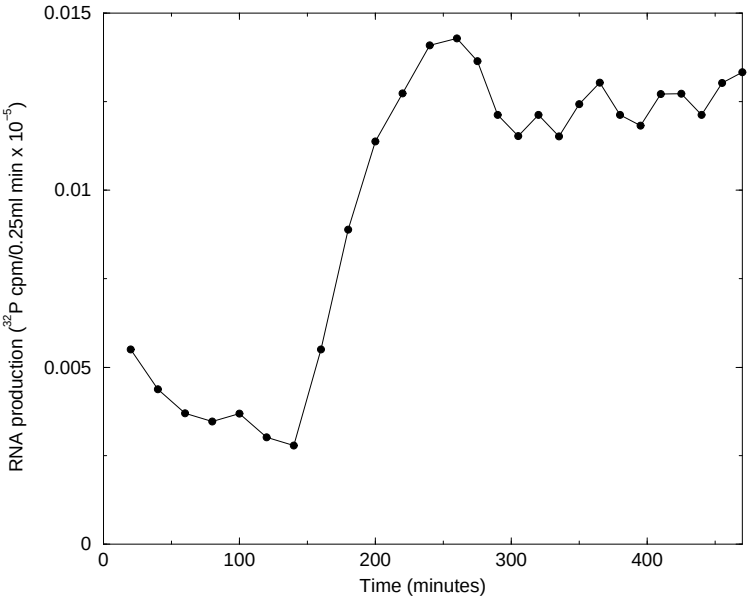



Figure 1.2: Increasing replication rate of evolving RNA molecules in a serial transfer experiment. Radioactive GTP is incorporated into newly synthesized RNA to measure the production [33].

generations the growth rate increased by about 50%. This fitness increase occurs in steps and not continuously. After this adaptive period the curve settles on a plateau at about 1.5 times of the initial fitness. The phenotypic evolution in terms of cell size or fitness was compared to genotypic changes [36]. Unlike phenotypic evolution, which is fast during the initial period, the genotypic evolution speeds up in the later saturation phase. Selection reduces diversity, neutral evolution increases it. These two counteracting tendencies are explored in section 3.10

## 1.2 Molecular Evolution

Molecular evolution *in silico* is built to follow Darwin's principles of variation and selection. Variation is a result of erroneous replication. Molecules are reproduced through copying of the ancestral molecules and inherit their sequence. Because of selection, fitter variants have a higher probability to

survive and to spread their genotypes to their descendants. Fitness values are assigned to genotypes in two steps. The first step is the mapping of genotypes to phenotypes. The genotype space $\mathcal{I}$ has a natural metric with the Hamming distance $d_{ij}^h$ between the sequences $i$ and $j$ [22]. It specifies the number of positions at which this two sequences differ. The phenotype space $\mathcal{S}$ can also be assumed as a metric space by defining a distance measure $d_{ij}^s$, the Hamming distance of shapes in parenthesis notation. We use it even though such a simple measure is far less natural, because this annotation of distance does not reflect the accessibility through Darwinian evolution, which is based on mutations on sequence level.

$$\psi : \{\mathcal{I}; d_{ij}^h\} \Rightarrow \{\mathcal{S}; d_{ij}^s\}$$

This mapping can not be expressed in analytical terms. At best there exists an algorithm, that assigns a phenotype $S_k$ to every genotype $I_k$. This can also be modelled by random graphs.

The second mapping evaluates phenotypes and returns a fitness value out of the non negative real numbers. This can be a function of a distance measure to a given target shape or any other property of the structure like the minimum free energy, the deviation of suboptimal shapes or the kinetic folding properties.

$$f : \{\mathcal{S}; d_{ij}^s\} \Rightarrow \mathbb{R}$$

The optimization process of molecular evolution is successful even without full knowledge of the target's properties. A fitness value assigned to every sequence leads the path to the target structure if a fitness increase is possible with usual mutations.

In summary a fitness value of a genotype can be calculated by the phenotype-genotype mapping and the fitness function, the assignment of a function of a distance value to the phenotype: $f_k = f(S_k) = f\big(\psi(I_k)\big)$

## 1.3 The Flow Reactor *in silico*

A computer simulation of molecular evolution has several advantages: (i) It gives us the ability to follow every single molecule from the beginning to the end of its existence and (ii) to record all interactions with other items. (iii) All parameters can be easily controlled and (iv) what usually happens in decades can be shortened to hours or days. On the other hand most evolutionary simulations are far from reality: It is not yet possible (i) to model all crucial three-dimensional interactions on molecular level, (ii) to supply with enough (random access) memory to hold a realistic population size, and (iii) to provide the necessary computer power. But simplifications to the essential factors can lead to valuable results. RNA folding gives rise to the simplest currently known genotype-phenotype mapping. Since it is possible to predict the RNA secondary structure, it is an ideal model for simulating molecular evolution.

An application of chemical reaction kinetics to molecular evolution is the quasispecies theory conceived by Manfred Eigen [5], which has been extended and further developed [7–10, 47]. His approach was to derive the mechanism by which biological information is created. Populations of RNA or DNA sequences migrate through sequence space and gain information by variation and selection. These populations are metastable but have a structured distribution around a master sequence. They optimize mean fitness by exploring new environments and create biological information laid down in genotypes. A stochastic process can have (i) a predefined target, which forms an absorbing barrier and sets an end point or (ii) an open end without a given target or time limit.

The replication-mutation system consists of $n \times n$ different processes of RNA synthesis and $n$ degradation reactions. In the mutation matrix $Q = \{Q_{ij}; i, j = 1, \ldots, n\}$ identical indices denote error free replication, while different indices stand for mutations. The uniform error rate model, which is assumed in this work, implies that the probability of a mutation is independent of the nature of the base exchange. Further only point mutations are possible, while inserts, deletions, inversions, translocations, or crossing-overs

are restricted. Nevertheless any sequence, which is part of the sequence space can in principle be created from any RNA molecule in the reactor. The replication accuracy per base $q$ and the derived mutation rate $p = 1 - q$ defines the frequency $Q_{kj}$ of a (erroneous) replication of a polynucleotide chain with length $\ell$ with a Hamming distance $d_{kj}^h$ between the template and the produced sequence:

$$Q_{kj} = q^\ell \left(\frac{1-q}{q}\right)^{d_{kj}^h} = (1-p)^\ell \left(\frac{p}{1-p}\right)^{d_{kj}^h}$$

Every sequence $I_k$ of the population with a frequency $x_j$ has a fitness value $f_j$ which is tantamount to its production rate constant.

The excess production rate of a reactor population $\Phi(t) = \bar{f}(t) = \sum_{j=1}^n f_j\, x_j(t)$ is the average fitness of the reactor population. This leads us to the replication-mutation differential equation:

$$\frac{dx_k}{dt} \;=\; x_k \Big(Q_{kk} f_k(t) \;-\; \Phi(t)\Big) \;+\; \sum_{j=1, j\neq k}^{n} Q_{kj} f_j(t)\, x_j \;, \quad k = 1, \ldots, n \;.$$

The production of copies and mutants from the actual population is equal to the excess production leading to constant population size $\sum_{j=1}^n dx_j/dt = 0$.

$$\begin{aligned}
* &\xrightarrow{r(t)} A \\
\ell A + I_k &\xrightarrow{f_k([A])Q_{kk}} 2I_k \\
\ell A + I_j &\xrightarrow{f_j([A])Q_{kj}} I_k + I_j \\
A &\xrightarrow{r(t)} 0 \\
I_k &\xrightarrow{r(t)} 0
\end{aligned}$$

The inflow and outflow of monomers $A$ is regulated by the current flowrate $r(t) = \Phi(t)$, which buffers the support of the reactor with low molecular weight building material.

From the template $I_k$ and $\ell$ monomers an error free copy is reproduced dependent on the frequency $Q_{kk}$ and the fitness values $f_k$. From another template $I_j$ also a copy of $I_k$ is produced. This is dependent of the template's fitness value $f_j$ and the (usually much smaller) frequency $Q_{kj}$.

Excess templates $I_k$ are removed from the reactor with the current flow rate $r(t)$.

A detailed description of the Gillespie algorithm [18, 19] used in this work can be found in section 3.1

## 1.4 Genetic Algorithms

A genetic algorithm (GA) is an optimization procedure based on the Darwinian principle of survival of the fittest. A set of possible solutions for a given problem is encoded in memory units and processed in a computer program. Each solution needs a computed fitness value or another measurement for fitness. Based on its fitness, the items are stochastically selected and replicated. This can be a recombination of 2 items, an insertion, deletion, or a point-mutation of the encoded memory chunks [30]. The possible changes of the items are called the moveset and are important for the success of a GA. Within this moveset the concept of neighbourhood is generated. Crucial properties of the solutions have to be preserved but increase of fitness must be possible between neighbours.

The traveling salesman problem (TSP) for example, where a man has to visit $n$ cities exactly once and then return to the starting point. His goal is to find the shortest route through all $n$ cities. The computer memory representation for one solution can be an vector of integers, where every city is encoded with a number. The fitness evaluation sums the distances between consecutive cities in the vector. One easy moveset might be the exchange of two randomly chosen cities. Under a few hundred vectors, each representing a path through the cities, one solution of the TSP is selected for replication, based on the relative fitness, iterately. A few of them will be templates for fitter items bringing the best solution in the population closer to the global optimum. To keep the

size of the population constant, a randomly chosen solution is removed after every replication.

Genetic algorithms are used for problems with a huge space of possible solutions, that can never be fully calculated. It is based on the assumption, that better solutions can be found in the neighbourhood of good ones.

In the mid-1970 John Holland first presented the concept of genetic algorithms in his pioneering book [26]. Because of the probabilistic selection, the best individual is not necessarily selected for replication and the worst one can still remain in the population. But nevertheless better solutions are favoured in general. This gives GAs an advantage against pure hill climbing methods which often fail with nontrivial problems, because a found, local maximum causes the algorithm to terminate [39].

The procedure of finding fitter items in a genetic algorithm can be very different. All selected items can be replicated to form the next generation, while the rejected solutions are discarded. Another way is to take a portion of a population for several steps of replication until a certain number of items are created followed by another reduction step (serial transfer). The population size can also be controlled dependent on the given average population size by increasing or reducing the probability of replication or outflow, respectively. In tournament selection groups of two or more individuals are formed. The best item out of each of these groups is taken for replication while the others are removed.

## 1.5 RNA and Secondary Structure Prediction

In difference to DNA, the major portion of RNA is single stranded. Intramolecular back-folding forms nucleotide pairs between adenine-uracil, guanine-cytosine (Watson-Crick pairs), or guanine-uracil (wobble base pairs). The secondary structure of RNA molecules is a simplification of its three dimensional shape. With the restriction not to form pseudo knots, only base pairs are indicated, while other intra-molecular interactions are overlooked. It can be displayed as a series of parenthesis and dots, where the corresponding parenthe-

sis show the position of base pairs in the sequence. The secondary structure of an RNA sequence covers the major share of the free energy of the tertiary structure formation and can be predicted using folding algorithms based on thermodynamic data [25, 28, 57]. Beside the minimum free energy (mfe) structure, which is formed after sufficiently long time and adequate low temperature, the Boltzmann weighted suboptimal confirmations in the sense of a partition function can be calculated [55]. Another approach is kinetic folding, which considers available folding time and the RNA transcription from 5' to 3'-end [12].

## 1.6 Shape Space

There are four generic properties of folding, the prediction of the minimum free energy (mfe) secondary structure of an RNA sequence [46].

(i) There are by far **more sequences than structures**. While the number of sequences with chain length $\ell$ and a four letter alphabet (A, U, G, C) is given by $4^\ell$, the number of possible structures is limited. If you consider the parenthesis-dot notation (see section 3.1) only a three letter alphabet is used. More restrictive is the limitation not to consider pseudoknots. If the set $P$ of paired positions contains two base pairs (i,j) and (k,l) then $i < k < j$ implies $i < l < j$. Into the bargain comes a minimal stack and loopsize which gives an approximation for larger chains length:

$$N_S(\ell) \approx s(\ell) = 1.4848 \times \ell^{-3/2}(1.84892)^\ell$$

(ii) **Some secondary structures are more frequent than others.** In the limit of long chains the fraction of such structures tend to zero, while in the fraction of sequences folding into them tends to one [21]. A structure is frequent or common, if it is formed by more sequences than average [43]. Although the number of frequent structures grows exponentially with increase of chain length, it is getting a smaller share of all possible structures.

(iii) **Neutral Networks** are formed by sequences with the same mfe-structure being connected by a single point mutation. Groups of these nearest

neighbours, can form extended networks in sequence space. This allows structure neutral mutations, that can replace the whole primary sequence step by step without changing the structure. This property can be found with frequent structures only.

(iv) All common structures can be found in a small radius of any randomly choosen sequence **(shape space covering)**. For example for chain length 100 only 15 mutations are necessary to find every frequent structure [42, 44].

For RNA sequence with equal chain length a natural metric is given by the Hamming distance [22]. For structures the situation is far less clear. For three dimensional shapes the root mean square deviation is used. For RNA secondary structure the *tree edit distance* can be applied. But from an evolutionary point of view this measures are artificial because there is no physical process which modifies structure at this level of representation, but through mutation of its underlying sequence. The underlying sequences of two highly dissimilar structures can be almost identical or, on the other hand, similar or equal structures can be based on very different sequences. On the set of phenotypes we are concerned with a topology rather than a metric.

Consider the set $S_\alpha$ of all sequences folding into a given structure $\alpha$. The sequences with Hamming distance 1 to $S_\alpha$ are defined as $B_\alpha$, the 1-boundary of $\alpha$. The minimum free energy (mfe) structures $\Sigma_\alpha$ of the sequences $B_\alpha$ are the 1-accessible structures of $\alpha$. The notation for accessibility of a structure $\beta$ from $\alpha$ is $\beta \leftarrow \alpha$ and $\Sigma_\alpha = \{\beta | \beta \leftarrow \alpha\}$ can be defined.

Through inverse folding one can find a representative sample of $S_\alpha$ [25]. A ranking of the commonness of structures of $\Sigma_\alpha$ can be made based on two principles. The *neighbourhood frequency* $\nu(\beta, \alpha)$ counts the share of sequences of $S_\alpha$, in which boundary at least one copy of a structure $\beta$ can be found, while the *frequency of occurrence* $\vartheta(\beta, \alpha)$ counts also the share of occurrences in its boundary. For both frequency measurement we define the set

$$\Psi_\epsilon(\alpha) = \{\beta \in \Sigma_\alpha \mid \rho(\beta, \alpha) \geq \epsilon\} \ ,$$

where $\rho(\beta, \alpha)$ can be $\nu(\beta, \alpha)$ or $\vartheta(\beta, \alpha)$.

A given limit $\epsilon$ defines the top ranking structures as the *characteristic set*

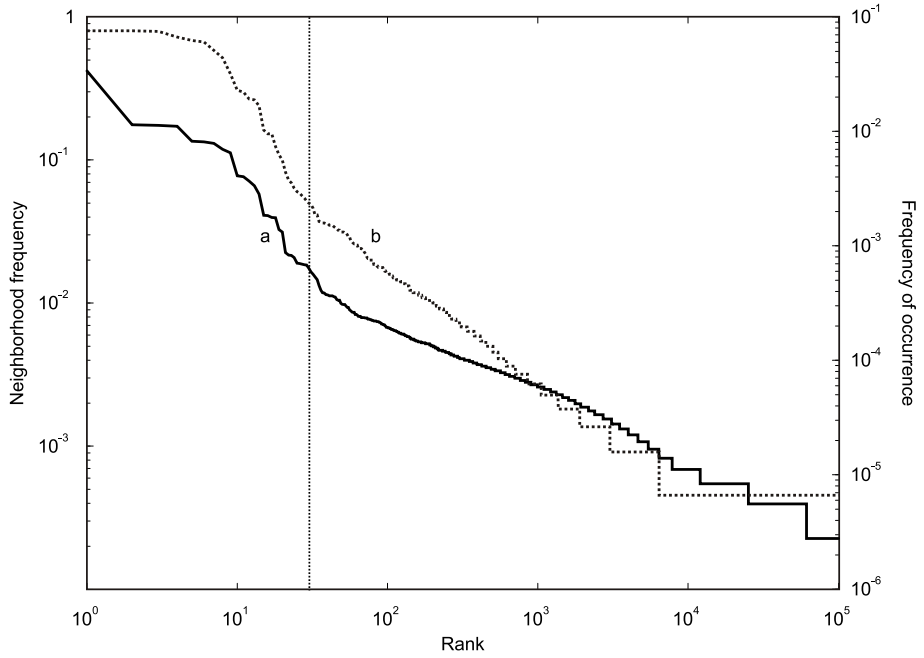



Figure 1.3: A log/log plot of the rank ordered structures in the boundary of $\mathrm{tRNA^{phe}}$. 28% of the neighbours of 2199 sequences folding into the clover-leaf structure formed the same shape than their reference sequence and thus belong to the neutral network. Curve **a** (right ordinate) shows the rank ordered frequency of occurrence $\vartheta(\beta, \alpha)$, while the neighbourhood frequency $\nu(\beta, \alpha)$ is plotted in curve **b** (left ordinate). The dotted vertical line seperates the frequent structures in the boundary of $\mathrm{tRNA^{phe}}$ (right) from the hardly reachable shapes on the left. This is typical for a scaling according to Zipf's law, which implies that the log(frequency)/log(rank)-plot is a straight line [56].

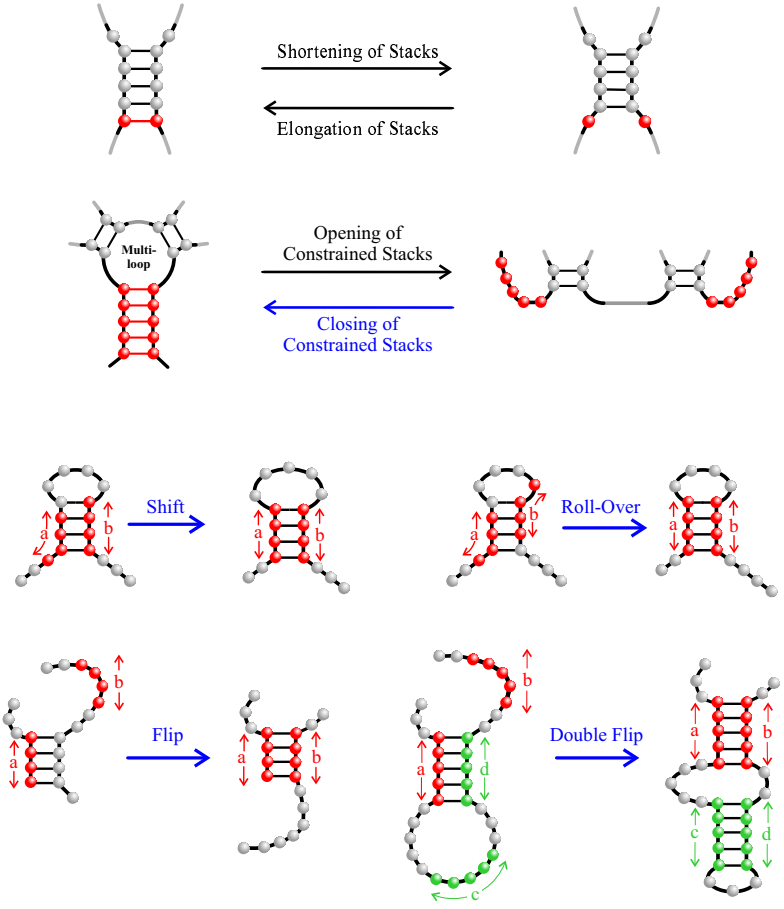



Figure 1.4: Continuous and discontinuous RNA transitions. Black and blue arrows show continuous and discontinuous transitions, respectively. On top the loss and formation of a base pair are both continuous transitions. The middle part sketches a one-way continuous transformation. The loss of a constrained stack is continuous, while the closing is discontinuous. On the bottom four different types of discontinuous transformation are shown, which are summarized as generalized shifts.

of $\alpha$. A shape $\beta$ is defined as near to $\alpha$ if $\beta$ is an element of $\Psi_\epsilon(\alpha)$. *Continuous transitions* are transitions to the *characteristic set* of a structure. This are minor changes like opening or closing a base pair at one end of a stack, while *discontinuous transitions* are major structural changes like a shift of a whole base pair stack and affect structures which are not in the set $\Psi_\epsilon(\alpha)$.

The nearness of $\alpha$ to $\beta$ needs not to be symetric. E.g. to open the closing stem of a multiloop shape $\alpha$ is mostly possible with a single mutation and is a *continuous transition.* But closing such a stem requires the unlikely coincidence of matching base pairs between the participating bases and is a *discontinuous transition.*

The *relay series* are a kind of relay-race on the level of structures. Starting from the found target shape a backtracking to the inital structure is made. First the shape $\gamma$ which gives rise to the target shape is explored. The former step in the *relay series* is the shape which gives rise to $\gamma$. This procedure is repeated until a shape of the inital population at the beginning of the flow reactor run is reached.

In flow reactor simulations the evolution of the mfe-structure of an arbitrarily chosen sequence to the given target shape occurs in fitness jumps. Longer periods of almost constant fitness are interrupted by phases of fast fitness gain. The quick fitness increase is called a major transition. The comparision of the fitness plot and the relay series show that major transitions are usually discontinuous.

## 1.7 Organisation of This Work

This work starts with a description of the programming class libraries for flow reactor simulation and analysis, which was developed during this thesis (chapter 2). In section 3.1 a description of the algorithm of the flow reactor simulations used in this work can be found. Section 3.2 is a statistical summary on evolutionary trajectories. In section 3.3 the distribution of the number of replications and replication times in relation to population sizes are described. Section 3.4 deals with the stochastic dynamics of neutral evolution. The sur-

vival probability of a single sequence in a flow reactor is examined numerically in section 3.5. A comparison of the performance of a flow reactor of this type to serial transfer experiment is the topic in section 3.6. The error threshold and its influence on population size is explained in section 3.7. With the flow reactor simulation the lineage of the evolution in a flow reactor can also be traced, which is described in section 3.8. A comparison of the lineage on the level of structures with the relay series can be found in section 3.9. The movement and the spreading of the reactor population in sequence space in comparison to shape space during this evolutionary process show some similarities to recently published *in vitro* experiments [36] (section 3.10). A graphical interpretation of the movement in sequence space is the topic of section 3.11. The formation and development of clusters during this evolutionary optimization process is described in section 3.12. A comprehensive model of evolution is presented in chapter 4. Finally in chapter 5 the results and future aspects are discussed.

# 2 Flow Reactor Class Library and Application

## 2.1 Main Components

The *Flow Reactor Class Library* is a programming library for evolutionary computation in simulated flow reactors. It is written in the programming language C++ and is based on the Standard Template Library (STL), which is part of the the ISO/ANSI 14882:1998(E) approved standardization of this language [38, 41].

The core part is the *reactor class*. It manages the sequence objects defined in the *sequence class*, selects them for replication or outflow, advances the internal clock and writes the log-files in order to analyse the reactor runs in retrospect. Two different flow reactor algorithms are provided: (i) The Gillespie algorithm [18, 19] for continuous flow reactors is explained in section 3.1 while (ii) the serial transfer algorithm, which simulates a batch reactor is described in section 3.6.

From the *reactor class* several classes are derived, which

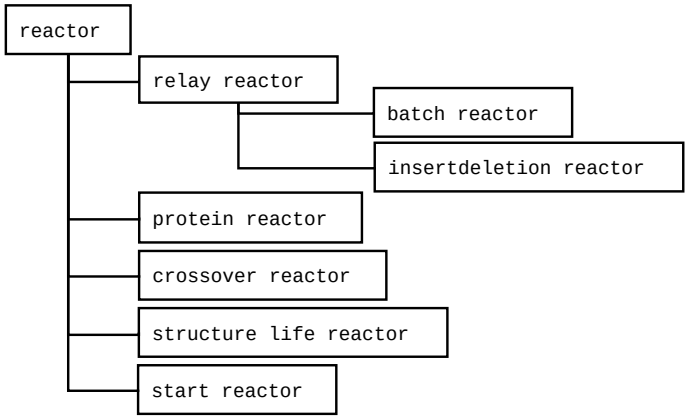



Figure 2.1: The hierarchy of the reactor class and its descendants.

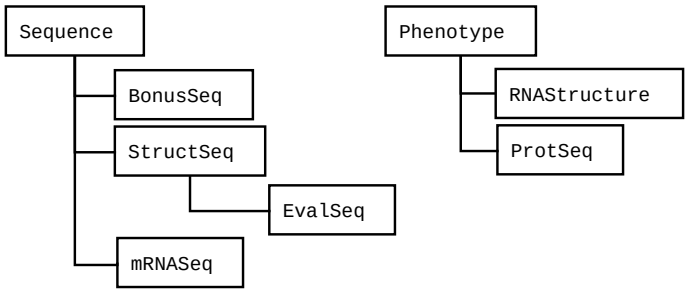



Figure 2.2: The hierarchy of the sequence class as well as the phenotype class and its descendants.

- log the production and extinction of phenotypes (*relay reactor*)
- contain simulated mRNA, which is translated into proteins and calculate its fitness by similarity to another protein in terms of primary structure or shape similarity (*protein reactor*)
- mutate its sequences by insertions and deletions (*insert-deletion reactor*)
- mutate its sequences by crossovers (*crossover reactor*)
- terminate after certain events (*structure life reactor*, *start reactor*)
- use the serial transfer algorithm (*batch reactor*)

The *sequence class* and its descendants, which are managed by the reactor classes handle RNA sequences and calculate their fitness values, which can be based on very different criteria: (i) As described in section 1.2 this can be a function based on the mfe-structure, or (ii) the minimum free energy, (iii) the free energy of the sequence, folded into a given shape, (iv) the similarity to other proteins, after translation of the RNA into a protein sequence or any other appropriate function. In many cases the library functions of the *Vienna RNA Package* [25, 52] are used.

In some cases the sequences, which represent the genotype are linked to the *phenotype class* and its descendants. They represent phenotypic properties like the translated protein sequence or the secondary structure of RNA. Their properties are inquired by the *sequence class* for its fitness calculation. These classes are also important for the creation of relay series and lineages of a flow reactor run. They keep all structures in memory appearing during the

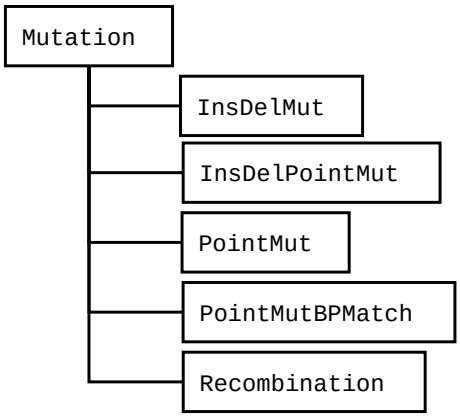



Figure 2.3: The hierarchy of the mutate class and its descendants. Point mutations, base pair aware point mutations, insertions, deletions, and recombinations are possible with these classes.

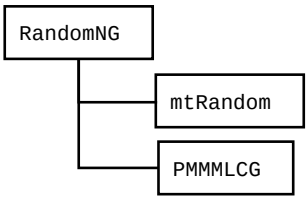



Figure 2.4: The hierarchy of the random number generator class and its descendants. The base class *RandomNG* is an abstract class, from which the *Mersenne Twister Random Number Generator (mtrandom)* [32] and the *Prime Modulus M Multiplicative Linear Congruential Generator (PMMMLCG)* [37] are derived.

flow reactor run to allow efficient searches for former appearances and to keep consistent numbering of phenotypes.

The mutation classes *(mutation, insdelmut, insdelpointmut, pointmut, pointmutbpmatch)* provide all kinds of mutation, like point mutations, insertions, deletions, and special changes like base pair aware mutations. With a special class written by Jörg Hackermüller also recombinations are possible.

Finally the *randomwell class* is an envelope for the request to the pseudo random number generator classes (*randomNG, mtRandom* and *PMMMLCG*), which provide the application of two pseudo random number generator algorithms, the *Mersenne Twister Random Number Generator* [32] and the *Prime*

*Modulus M Multiplicative Linear Congruential Generator* [37].

The *Flow Reactor Class Library* allows users to execute evolutionary simulations and to easily reproduce the essential results presented here. Since these programs are written in C++ it is easy to extend functionality without changing or copying the current code. The main components, (i) the reactor classes, the (ii) sequence and (iii) phenotype classes as well as the (iv) mutation classes, cover the main functionality modules which are separated from each other. If one day the computer capacity and algorithms are available to calculate tertiary interactions of sequences within one second of time, classes from sequence class and phenotype class can be derived without touching the reactor or mutation functionality. For all results in this thesis only standard i386 hardware and the linux operating system was used (the programs compile on other commercially oriented operating systems also).

## 2.2 Analysis I

The logging of a flow reactor run is written in at least four file types:

- The columns of the monitor file (tst.mon) denote (i) the current time, (ii) the number of sequence innovations, (iii) the number of replications, (iv) the current capacity, (v) the id of the dominating sequence and (vi) its current population size, (vii) the quotient of different sequences and the population size (=sequence diversity), (viii) the average rate constant, (ix) the average distance to the target shape and (x) an indicator if the target shape is already present in the population (1..true, 0..false).

- The history file (tst.history) contains the creation time of every structure, its parent shape number as well as the extinction time, Hamming distance to the parent sequence and the parent shape frequency at the given time.

- The dumps of all sequences (tst0, tst1, ..., tstn) and its ID, fitness and population size is usually written every 5 time units.

- At the end of every run a file of all structures (phenotypes) and its IDs is created.

For the analysis several Perl scripts [51] were created, which give scientists the ability to explore details of the flow reactor runs:

A script for the creation of the relay series [15, 16, 45] is provided *(relay.pl)*, which reads (i) the history file and (ii) the file of all structures (phenotypes). Its output is a summary file *(traj.dat)*, which shows among other things (i) the transition time $t_{trans}$ of structures in the relay series, (ii) the Hamming distance of the sequences whose shapes made the transition, (iii) the population size of the ancestor shape at $t_{trans}$ and (iv) the shape itself in parenthesis notation. Further a list of life spans of all structures involved in the relay series *(life.dat)*, and a file giving the relay step number and the transition time $t_{trans}$. These data are sufficient to create images like figure 3.2.

Other scripts show the length of the start phase *(fitjump.pl)* and differ between continuous and discontinuous transitions *(discont.pl)*.

The births, deaths and parentships of every structure can be examined with *achievement.pl*.

## 2.3 Analysis II

For further analysis of flow reactor runs the *Flow Reactor Analysis Class Library* written in C++ is provided. The core components are the *sequence* class, which denote an RNA sequence or an RNA structure in parenthesis notation, and the *position* class, which enable us to analyse RNA sequence positions, like the frequency of occurrence of a nucleotide.

The data of a bunch of sequences, like a dump of a flow reactor at a certain time, can be analyzed on the level of sequences or on the level of positions: The class *group of sequences* is provided to make comparisons of several sequences, like the average Hamming distance between two dumps (see equation 3.29). The derived *distance cluster* class calculates the subgraph clustering discussed in section 3.12 using the struct *edge*. The *group of positions* class keeps the

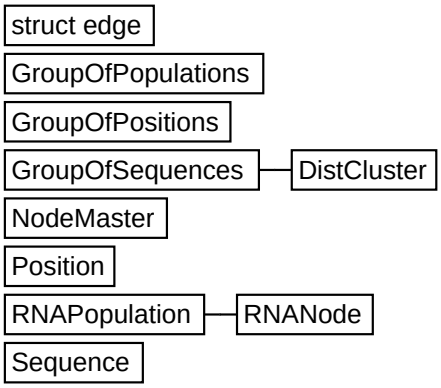



Figure 2.5: The hierarchy of the classes of the *Flow Reactor Analysis Class Library* for the analysis of flow reactor runs.

number of different bases on every position after reading a sequence dump. With this data the distance between mean nucleotide sequences (see equation 3.30) or the consensus sequences can be easily calculated.

In many cases both, the sequence data and the position data are needed. This is provided with the *RNAPopulation* class, which is utilized for the calculation of the variance/covariance matrix (see section 3.11) or the standard deviation. Derived are the *RNA Node* class for the creation of Ward's minimum variance method clusters [53] explained in section 3.12. These clusters objects are collected in an object of class *NodeMaster*.

# 3 Numerical Results

## 3.1 *In Silico* Flow Reactors

The flow reactor simulations in this work are based on the algorithm developed by Daniel Gillespie [18, 19]. At the beginning the reactor contains the inital population of RNA sequences with equal chain length. The goal of the reactor run is to find a sequence, whose minimum free energy (mfe) structure is a given target shape. Only two types of reactions, (i) the replication of an RNA molecule and (ii) its outflow from the reactor, are possible.

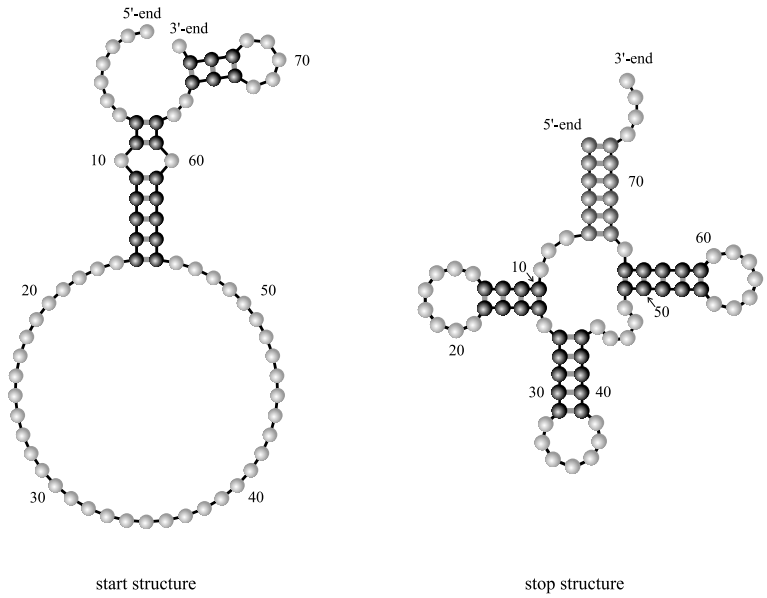
The rate constant, which is proportional to the selection probability of a reaction, must be calclulated for every sequence and reaction type. The term sequence is used as a shorthand for a RNA sequence type with a distinct order of nucleotides. For every sequence, which is present in the reactor, the actual number of copies $n_i$ can be determined.

RNA secondary structures can be represented by strings written in a shorthand notation using parenthesis and dots. Parenthesis correspond to bases combined to base pairs, dots represent single bases. For example, the string of a typical hairpin loop reads:

```
..((((....)))).
abcdefghijklmno
```

The bases `c` and `n`, `d` and `m`, `e` and `l` as well as `f` and `k` form a base pair, respectively. All other bases are unbound.

The fitness value $f_i$ of a sequence with $\ell$ nucleotides is based on its mfe structure [25]. A structure distance to the given target shape can be the Hamming distance $d^s_{ij}$ [22], the tree edit distance $d^t_{ij}$ or the string edit distance $d^{st}_{ij}$ [13]. Unless otherwise stated the Hamming distance is used in this work (See also section 1.2).




```
AGUAAUUUGUAGCCAACACACCGCACUUCCUUGUCUUAGUUUUAACCUUCUAACUGGCUCCGUAGUUCUUGUAACG
.......((.(((((.....................................)))))..))..(((.....))).

((((((...((((........)))).(((((.......)))))).....(((((.......)))))..)))))....
```

Figure 3.1: The start and stop structure of the flow reactor runs. The initial population was chosen to be homogeneous, and thus all molecules have the sequence shown above. Below the sequence start and stop structure are shown in parenthesis notation. The distance between the two structures expressed as Hamming distance between the two strings is $d^s_{0,\tau} = 48$.

$$f_i = \frac{1}{0.01 + d_{ij}^s/\ell} \tag{3.1}$$

The rate constant for a replication reaction $r_i^{rep}$ of sequence $i$ is based on its fitness $f_i$ and its number of copies $n_i$ in the reactor.

$$r_i^{rep} = f_i \cdot n_i \tag{3.2}$$

While the probability of a replication is dependent on the fitness value of the RNA sequence $i$ and its number of copies $n_i$, the probability for the outflow reaction is only proportional to its population size $n_i$. The rate constant for the outflow reaction $r_i^{out}$ of a sequence $i$ is based on a given average population size $N_{set}$ and the sum $R(t)$ over the replication rate constants of all $k$ sequences at time $t$.

$$R(t) = \sum_{j=1}^{k} r_j^{rep} \tag{3.3}$$

$$r_i^{out} = R(t) \cdot \frac{n_i}{N_{set}} \tag{3.4}$$

$$N(t) = \sum_{j=1}^{k} n_j \tag{3.5}$$

If at time $t$ the population size $N(t) = N_{set}$, the probability for replication and outflow reaction is equal. If $N(t)$ is larger or smaller than $N_{set}$ an outflow or a replication reaction is favoured, respectively. In this way the population size is kept nearly constant and fluctuates around $N_{set}$ with a standard deviation of $\sqrt{N_{set}}$.

$$A(t) = R(t) + \sum_{j=1}^{k} r_j^{out} = R(t) \cdot \left(1 + \frac{N(t)}{N_{set}}\right) \tag{3.6}$$

The current reactivity $A(t)$ gives the interval $]0..A(t)]$ for the pseudo random number, which selects a replication or outflow reaction for the next step. If an outflow channel is selected a copy of the corresponding sequence is removed from the reactor population. By using a replication channel, a sequence

is copied base per base with a given accuracy (usually 0.999), which sometimes leads to point mutations but guarantees equal chain length.

$$\Delta t = \frac{\log\left(1/m_{prn}\right)}{A(t)} \tag{3.7}$$

The internal clock is advanced by $\Delta t$ using another **p**seudo **r**andom **n**umber $m_{prn}$ from the interval ]0..1] and the current reactivity $A(t)$. During the inital phase, when the fitness values and consequently the current reactivity are relatively small, time moves faster, while in a flow reactor with a higher population size or in a state almost before the target is reached it elapses much slower.

During a flow reactor run every creation and extinction of a shape or even of a sequence is recorded and kept in a log file, which enables us to track back the life of every structure or sequence.

## 3.2 Statistics on Evolutionary Trajectories

The **relay series** [15] is a way to reconstruct the succession of structures in an evolutionary process, simulated in the flow reactor. It is a list of structures beginning with the target shape and ending with a structure of the initial population.

Usually every structure in the reactor has multiple intervals of existence delimited by the shape's entrance and exit times. The relay series can be reconstructed easily only in retrospect, searching the log file for the shape $\alpha_{n-1}$, which gave rise to the target shape $\alpha_n$ when in finally appeared. Next the shape $\alpha_{n-2}$, which started the live interval of $\alpha_{n-1}$, during which $\alpha_n$ was produced, has to be found. These steps are repeated until a shape $\alpha_0$ is reached, that came into existence at the start of the reactor run as part of the initial population.

In every flow reactor run there are two distinct phases. At first there is (i) the initial period, where fast increase of mean fitness and many structural changes can be observed. Since the structure of the start population is usually very different from the target, many mutations with structural changes, lead

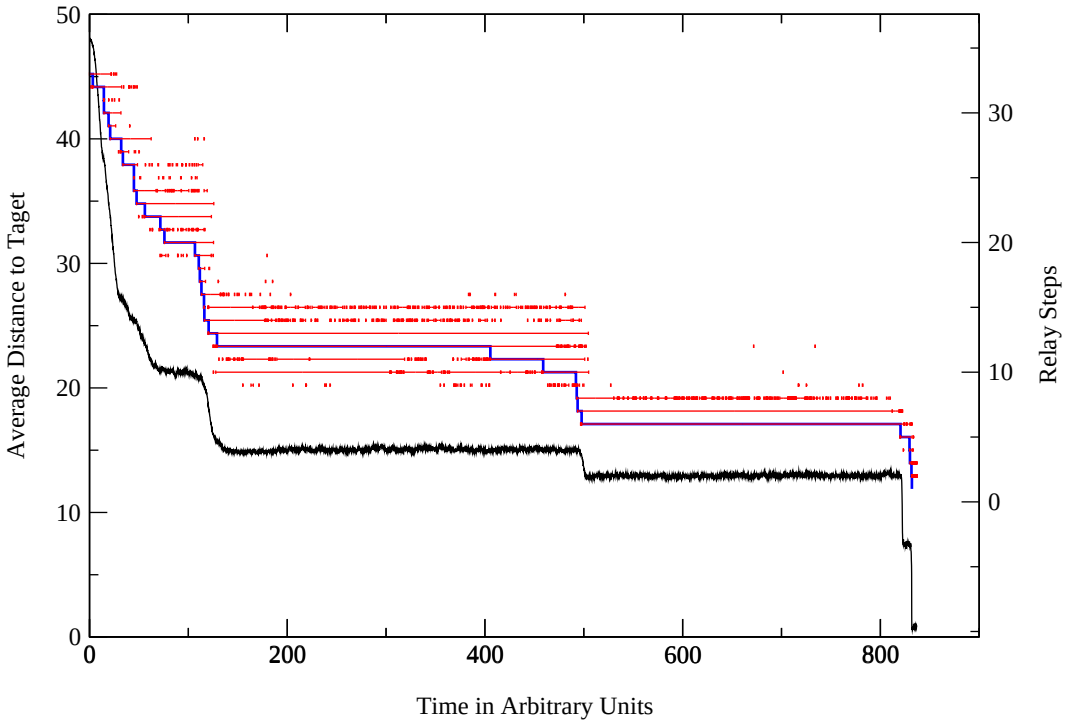



Figure 3.2: Evolution in a flow reactor with an average population size $N_{set} = 3\,000$ and a mutation rate per base and replication of $p = 0.001$. The start population and stop structure is show in figure 3.1. The black line shows the average Hamming distance of all sequence copies in the reactor to the target shape (left ordinate), which decreases stepwise. The red bars show the life spans of all structures involved in the relay series. The steps of the relay series are indicated by a blue line. The numbering of these steps is denoted on the right ordinate.

Table 3.1: The relay series of a flow reactor run with a average population size $N_{set} = 30\,000$. From the target structure (No. 1) a backtrack was made until the start structure (No. 21) was found (see text for details). The given Hamming distance denotes the distance between the sequences of shapes no. i+1 and i at which the relay transitions happened. Pop. size is the population number of shape no. i+1 at the time it produced i. Since shape no. 21 denotes the start population both of these values are undefined. c/d asterisks continuous and discontinuous transitions. The column "number of parents" counts the different shapes, that produced shape No. i. Except for the first production of every shape this is mainly back flow.

| No. | Production time | Hamming distance | Pop. size | No of parents | shape | c/d |
|---|---|---|---|---|---|---|
| 1 | 710.08 | 1 | 1 | 1 | ((((((...((((........)))).(((((.......)))))).....((((.......)))))..))))))....  | d |
| 2 | 710.00 | 1 | 15756 | 8 | ((((((...((((........((((.(((((.......)))))...))))..........))))..))))))....  | c |
| 3 | 639.93 | 1 | 797 | 2 | ((((((...((((........)))).(((((.......)))))).....((((((......))))))))))))....  | c |
| 4 | 639.66 | 1 | 341 | 1 | ((((((...((((........)))).(((((.......)))))).....((((.....)))))..))))))....  | d |
| 5 | 638.38 | 1 | 649 | 4 | ((((.....((((........)))).(((((.......)))))...((.(((((......)))))..))))))....  | d |
| 6 | 590.34 | 1 | 183 | 3 | (....)...((((........)))).(((((.......)))))..(((((((((.....))))..)))))).....  | c |
| 7 | 584.75 | 2 | 1129 | 2 | (....)...((((........)))).(((((.......)))))..(((((((((.....))))..)))))).....  | c |
| 8 | 576.85 | 1 | 1300 | 1 | (....)...((((........)))).(((((.......)))))..(((((((((.....)))))..)))))).....  | c |
| 9 | 145.28 | 1 | 945 | 1 | ((((((...((((........))))..........))))))..(((((((((.....)))))..)))))).....  | c |
| 10 | 138.41 | 1 | 46 | 6 | ((((((...((((........))))..........))))))..(((((((.......))))..)))))).....  | c |
| 11 | 131.69 | 1 | 10 | 9 | ((((((..((((........))))).........))))))..(((((((.......))))..)))))).....  | c |
| 12 | 131.05 | 1 | 3 | 11 | ((((((..((.((........)).)).........))))))..(((((((.......))))..)))))).....  | c |
| 13 | 130.59 | 1 | 8311 | 16 | ((((((.(((((.................)))))..))))))..(((((((.......))))..)))))).....  | c |
| 14 | 43.82 | 2 | 5728 | 1 | ((((((..(((((........)))..)).......))))))..(((((((.......))))..)))))).....  | d |
| 15 | 24.95 | 1 | 111 | 2 | (((((.....(((........))).)))))...........(((((((.......))))..)))))).....  | c |
| 16 | 19.70 | 1 | 4 | 4 | (((((.....(((........))).)))))......(((...(((((((.......))))..)))))))).  | d |
| 17 | 17.85 | 1 | 45 | 3 | (((((.....(((........))).)))))......((....))................((((...)))).  | d |
| 18 | 15.14 | 1 | 261 | 2 | (((((.....(((........))).)))))......(((.(((....((((........))))...))).))).  | c |
| 19 | 6.47 | 1 | 1414 | 2 | (((((.....(((........))).)))))......((....))....((((((......))))))........  | d |
| 20 | 0.04 | 1 | 29863 | 2 | (((((((.(.(((((.........................................)))))..)..)))))))......  | d |
| 21 | 0.00 | - | - | 8 | .......((.(((((.........................................)))))..))..(((.....))).  | - |

to higher fitness values. These new sequences supplant the actual population, before being supplanted itself. In contrast to the phenotypic[1] changes, the genotypic changes are small in this phase (see also section 3.10). The start of (ii) the second period is defined as the beginning of $\Delta t \geq 10$ time units without the creation of a structure, that causes a fitness innovation and remains in the reactor population to steady this new fitness level. In this second phase, usually the major fraction of the population has the highest currently assigned fitness value (between 80% and 95% at an error rate of 0.001 per base and replication). Long epochs of statis are interrupted by short but strong improvements in this period, which are called major transitions. With every major transition and the concurrent fitness increase the number of possible shapes with higher fitness values decreases. In many cases they can not be found in the neighbourhood of the dominating shapes.

The relay series consists of (i) continuous and (ii) discontinuous transitions. Diagnostics of discontinuous transitions are based on three criteria: (i) The newly created structure has never been present in the reactor, (ii) they involve major structural changes (see also section 1.6), and (iii) they are infrequent in the sense that they have globally a small probability to occur.

To distinguish minor and major structural changes three distance measurements between structures are used. (i) The *base pair distance* $d_{ij}^{bp}$ counts the minimal number of openings and closings of base pairs to transform one structure into the other. (ii) The asymmetric *base pair preserve distance* $d_{ij}^{bpp}$ counts the number of base pairs, that can not be found in the other structure. If $d_{ij}^{bpp} = 0$ and $d_{ij}^{bp} > 0$, new base pairs have been formed. The formation of a single base pair is always an extension of an existing stack, which occurs frequently. (iii) The *Hamming distance* $d_{ij}^{s}$ [22] , which is also needed for the detection, defines a lower limit for the base pair distance: $2 \cdot d_{ij}^{bp} \geq d_{ij}^{s}$. If $2 \cdot d_{ij}^{bp} > d_{ij}^{s}$, bases involved in base pairs are bound to other bases after the transition, which is always a major structural change and is summarized as *generalized shift* in [16]. If one of the following criteria is true, we are concerned

[1]The RNA base pair structure with minimum free energy, calculated with the Vienna RNA Package, is tantamount with the RNA phenotype in this work.

Table 3.2: Statistics of evolutionary trajectories. Different trajectories of *in silico* evolution towards a tRNA target, $S_\tau = S_{\mathrm{tRNA^{phe}}}$, were recorded for different population sizes $N_{set}$. Standard deviations refer to best fits of normal distributions.

| No. of runs | Population size $N_{set}$ | Major transitions | Min. distance at start of phase 2 | Relay steps phase 1 | Total no. of relay steps |
|---|---|---|---|---|---|
| 20 | 1 000 | $9.0 \pm 2.2$ | $24.1 \pm 4.0$ | $4.1 \pm 2.1$ | $114.1 \pm 88.5$ |
| 20 | 2 000 | $9.6 \pm 2.3$ | $22.4 \pm 4.2$ | $5.1 \pm 2.5$ | $62.8 \pm 25.6$ |
| 20 | 3 000 | $8.4 \pm 1.9$ | $21.0 \pm 2.1$ | $5.3 \pm 1.9$ | $49.1 \pm 23.5$ |
| 20 | 10 000 | $10.6 \pm 2.2$ | $18.6 \pm 2.1$ | $7.0 \pm 2.3$ | $37.1 \pm 11.0$ |
| 20 | 20 000 | $9.2 \pm 2.2$ | $17.7 \pm 2.3$ | $6.1 \pm 2.1$ | $29.2 \pm 5.0$ |
| 20 | 30 000 | $8.9 \pm 2.1$ | $16.9 \pm 1.9$ | $6.8 \pm 2.0$ | $30.5 \pm 6.7$ |
| 4 | 100 000 | $8.8 \pm 1.6$ | $16.8 \pm 0.8$ | $6.8 \pm 1.1$ | $24.3 \pm 4.7$ |

with a major structural change:

$$(i) \qquad 2 \cdot d_{ij}^{bp} > d_{ij}^{s} \tag{3.8}$$

$$(ii) \quad 2 \cdot d_{ij}^{bp} = d_{ij}^{s} \geq 4 \; and \; d_{ij}^{bpp} = 0 \tag{3.9}$$

Note that the type of transition between structures is also not symmetric. While the closing of a stem ($d_{ij}^{bpp} = 0$) is a very rare event, the opening ($d_{ij}^{bpp} > 0$) happens frequently and can never be a discontinuous transition.

The number of relay steps shows vast scatter but decreases considerably with increasing population size $N_{set}$. Although a strong decrease in relay steps with increasing population size $N_{set}$ can be observed, the number of discontinuous transitions stays almost constant.

Smaller flow reactors provide too small space to hold many different shapes, that could connect two discontinuous transitions in a relay series. These shapes have to be recreated again which leads to some extra continuous relay steps. But a minimal number of continuous relay steps will sustain even in reactors with population sizes much larger than in our experiments. After a discontinuous transition, which is associated with a fitness increase in many cases,

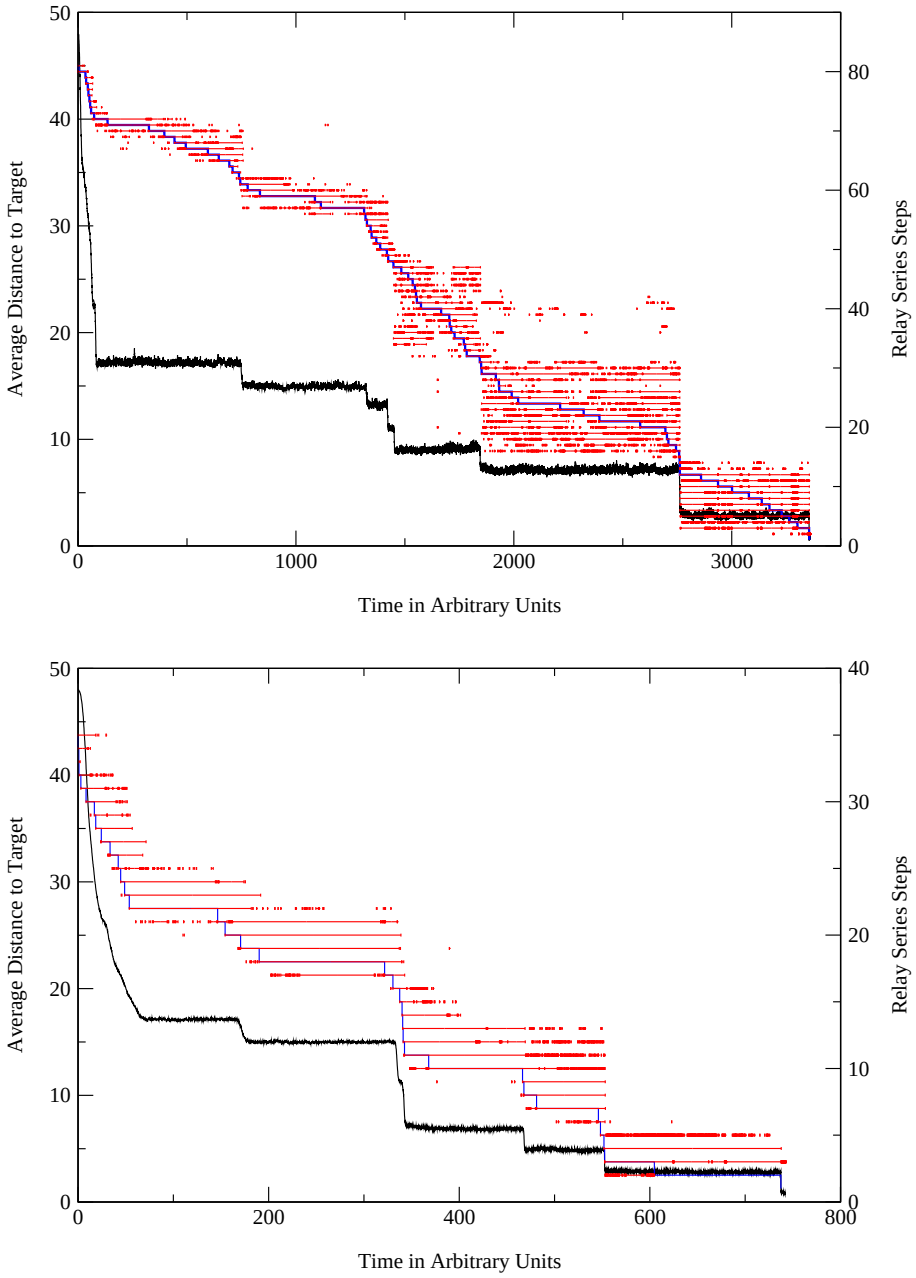



Figure 3.3: The relay series and average Hamming distance to the target shape (black line) of two flow reactor runs with a population size of 1 000 (above) and 10 000 (below), respectively. The relay series (blue line) of the run with $N_{set} = 1\,000$ consists of 80 steps (9 of them are discontinuous) and 66 different shapes. On average each of these shapes has 128 life spans (red lines, delimited by small red vertical lines). The flow reactor run with $N_{set} = 10\,000$ needs only 34 relay steps (11 are discontinuous). All of these 35 structures are unique and have 47 life spans on average.

a single structure $\beta$ with fitness $f_\beta$ becomes the ancestor of all future structures. New shapes on the same fitness level are created through continuous transitions and established in the population. The example in figure 3.4 shows, that a few relay steps are sufficient, if many different structures are present without interruption. In smaller reactors the population size $N_{set}$ seems to be too small, to make survival likely for many new created shapes on fitness level $f_\beta$. Whereas with increasing population size, the probability rises, that many or all crucial shapes stay present in the reactor until the next fitness gain takes place.

On a fitness plateau the major part of the reactor population has maximal fitness. The continuous creation of a shape through accurate replications, mutations on the same net, and inflow through mutations of sequences with other shapes has to compensate the outflow reaction to ensure the survival of that structure. To increase the number of mutated sequences, which are on the same neutral net, a drift into more connected regions seems to be a solution [34]. With the current setting this phenomenon can not be observed. If a higher mutation rate is used flat regions in sequence space are preferred [54].

## 3.3 Replications and Replication Time Distribution Statistics

Most frequency distributions in life are skewed and fit the log-normal distribution. Very well known is the distribution of income, which evidently is not symmetric around the mean, but much extending to the right [17]. The distribution of replications of RNA sequences to reach a given target shape in a flow reactor is also no symmetric distribution. The normal distribution doesn't apply, because the probability density function is right skewed and the standard deviation is so high, that the confidence interval includes values below zero. Although we have to few results to perform a test of deviation, the log-normal deviation (with basis $e$) is presumed.

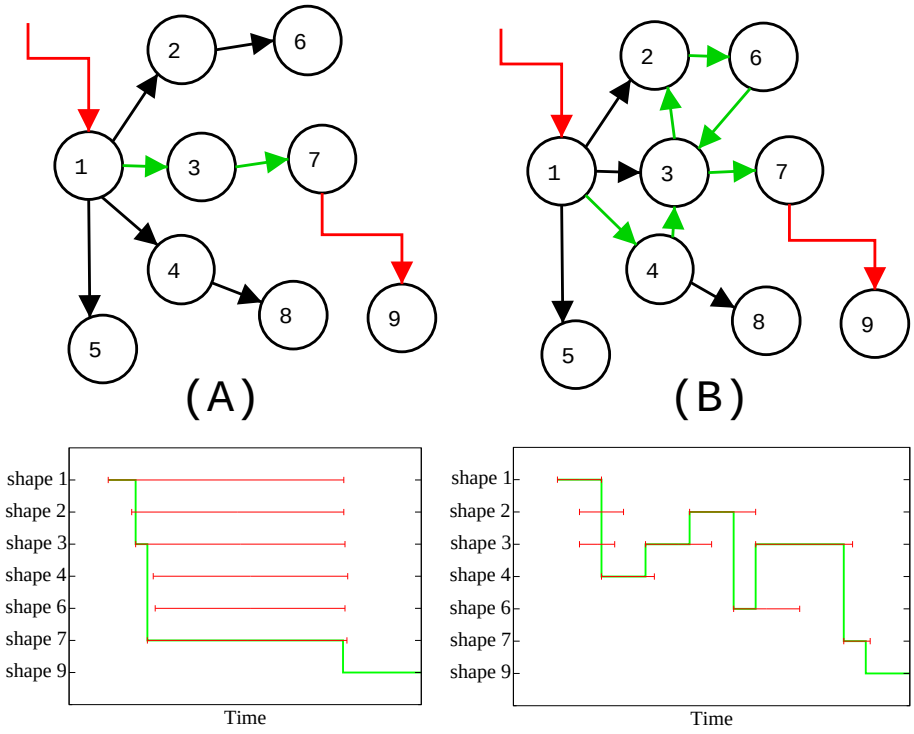



Figure 3.4: A simplified representation of the different number of relay steps during continuous transitions in large (**A**) and small (**B**) flow reactors, respectively. The circles denote shapes, transitions are represented by arrows. Shape 1 as well as shape 9 were created through discontinuous transitions (red arrows), which are the entrance and exit points in this examples. All other transitions are continuous. In the lower part of the figure the relay series (green line) and the life spans of shapes (red bars) are shown. **A** shows a scenario in a reactor with a large population size. The newly created structures (1-8) stay present until the final discontinuous transition produce a fitter structure, that supplants them. Only few relay steps are necessary between the two discontinuous transitions ($1 \rightarrow 3 \rightarrow 7 \rightarrow 9$). Whereas in a small reactor in part **B** new, but fit shapes die out and have to be recreated frequently. 7 relay steps are needed to link shape 1 and 9 ($1 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 6 \rightarrow 3 \rightarrow 7 \rightarrow 9$).

Table 3.3: The population size and the $log_e$ of the average number of replications $\bar{\nu}_{rep}$ to reach the target shape. A significant increase with increasing population size can be determined.

| No. of runs $m$ | Population size $N_{set}$ | $\bar{\nu}_{rep}$ |
|---|---|---|
| 20 | 1 000 | $17.8 \pm 0.79$ |
| 20 | 2 000 | $17.9 \pm 1.02$ |
| 20 | 3 000 | $18.0 \pm 1.39$ |
| 20 | 10 000 | $18.6 \pm 0.70$ |
| 20 | 20 000 | $18.8 \pm 0.62$ |
| 20 | 30 000 | $19.2 \pm 0.85$ |
| 4 | 100 000 | $19.6 \pm 0.39$ |

$$\nu_{rep}^k = log_e(n_{rep}^k) \tag{3.10}$$

$$\bar{\nu}_{rep} = \frac{1}{m} \cdot \sum_{k=1}^{m} \nu_{rep}^k \tag{3.11}$$

For convenience the numbers of replication $n_{rep}$ are logarithmically transformed and treated as normal distributed. The results for population size 100 000 were eliminated, because there are too few values available to obtain expressive statements. With the 120 results from population size 1 000 to 30 000 the analysis of variance (ANOVA) results in a significant difference (p-value $= 8.04 \times 10^{-6}$), which shows that the number of replications to reach the target is **not** independent on the population size of the reactor.

| Reason of Scattering | SS | DF | MS | F | P-Value | crit. F |
|---|---|---|---|---|---|---|
| Diff. between Groups | 32.475 | 5 | 6.495 | 7.109 | $8.043 \cdot 10^{-6}$ | 2.294 |
| Diff. within Groups | 104.155 | 114 | 0.914 | | | |
| Total | 136.630 | 119 | | | | |

SS..sum of squares, DF..degrees of freedom, MS..mean square, F..F-value

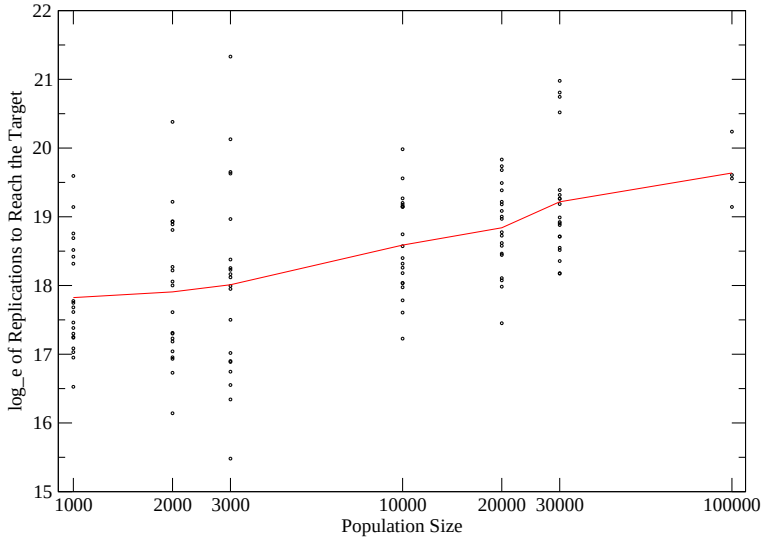



Figure 3.5: The population size vs. the number of replications to reach the target show a log-linear correlation and the assignment of the values to the right population size, can be distinguished from chance. The red line connects the arithmetic means $\bar{\nu}_{rep}$.

One of the reasons for the worse performance of big reactors might be the fact, that the replacement of the whole population after a fitness innovation, takes much more replications, than in a small reactor. If in a discontinuous transition a fitter, but hardly reachable structure is found and it survives, it will be the single ancestor for the whole future population. But first inferior sequences, which still exist in the reactor, amount to the major fraction of replication rate constants and are therefore still replicated, before they get supplanted by the fitter sequences. In more separated fitness landscapes, this "waste" of replications increase the chance, to reach other fitness islands in sequence space, but in a highly connected neutral network of a frequent structure, it might be one of the reasons for a higher average replication number to reach the target.

The replications to reach the target shape scatters vastly. Finding a fitter structure is a rare event at in phase 2. The Hamming distance between shifted structures is relatively small and leads to small fitness differences between shifted structures (see figure 1.4). If a structure is found, which is only a shift away from the target, it is usually very time consuming to reach the target shape. Most of the bases of a sequence involved in that shift have to match both structures before a mutation results into the fitter shape. Since this nearly matching sequences don't compete better than any other structure on that fitness level, it is unlikely that they become present and reach a high population size. To wait for this unlikely event leads to a vast scatter in the number of replications to reach the target structure.

## 3.4 Stochastic Dynamic of Neutral Evolution

### 3.4.1 Transition Probability

Since we assume asexual replication without deletions or insertions only independent point mutations are possible. The changes of genotypes follow the uniform error rates model which implies that error rates do neither depend on the nature of the nucleotide exchange nor on the position in the sequence.

Then, the probability of $k$ point mutations in a single replication event are expressed by $\text{Prob}\{k \text{ mutations}\} = \binom{\ell}{k} \cdot p^k \cdot (1-p)^{\ell-k}$, where p is the error rate per site and replication and $\ell$ the chain length of the molecule. The term $(1-p)^{\ell-k}$ represents the probability that the remaining $(\ell - k)$ nucleotides of the molecule are replicated correctly [6]. We derive the probability of a *specific* single point mutation $I_i \to I_j$,

$$\text{Prob}\{I_i \to I_j\} = \frac{N_i}{N} \cdot \frac{p^k \cdot (1-p)^{\ell-k}}{\kappa - 1} = \xi_i \cdot \frac{p^k \cdot (1-p)^{\ell-k}}{\kappa - 1} = \xi_i \cdot P_{ij}\,, \quad (3.12)$$

with $N_i$ being the number of genotypes $I_i$ in a population of size $N = \sum_{i=1}^{n} N_i$, $\xi_i$ being its relative frequency, and $\kappa$ the size of the nucleotide alphabet. The transition matrix, $\mathbf{P} = \{P_{ij}; i, j = 1, \ldots, n\}$, is symmetric as a consequence of the uniform error rate assumption. Because of the small mutation rate ($p = 0.001$) in the examples discussed here, it is a good approximation to restrict to single point mutations ($k = 1$). 99.73% of all replications of a tRNA molecule with $\ell = 76$ bases have a single or no mutation ($(1-p)^{76} + p \cdot (1-p)^{75} = 0.9973$).

The relations between the genotypes $I_\cdot$ and the associated phenotypes is modelled by the mapping $S_j = \psi(I_\cdot)$. Two classes of neutrality are considered: (i) Several genotypes are assumed to form the same phenotype, $S_j = \psi(I_i)$ $\forall\ i = n_{j-1} + 1, \cdots, n_j$

$$\underbrace{I_1, \ldots, I_{n_1}}_{S_1}\,,\ \underbrace{I_{n_1+1}, \ldots, I_{n_2}}_{S_2}\,,\ \underbrace{I_{n_2+1}, \ldots, I_{n_3}}_{S_3}\,,\ \cdots,\ \underbrace{I_{n_{j-1}+1}, \ldots, I_{n_j}}_{S_j}\,, \cdots$$

and (ii) several phenotypes may have indistinguishable fitness values, $f_k = f(S_\cdot)$.

$$\underbrace{S_1, \ldots, S_{n_1}}_{F_1}\,,\ \underbrace{S_{n_1+1}, \ldots, S_{n_2}}_{F_2}\,,\ \underbrace{S_{n_2+1}, \ldots, S_{n_3}}_{F_3}\,,\ \cdots,\ \underbrace{S_{n_{j-1}+1}, \ldots, S_{n_j}}_{F_j}\,, \cdots$$

The number of phenotypes $S_j$ is denoted by $M_j$. Summation over all genotypes yields $M_j = \sum_{i=n_{j-1}+1}^{n_j} N_i$ and $\sum_{j=1}^{m} M_j = \sum_{i=1}^{n} N_i = N$. In order

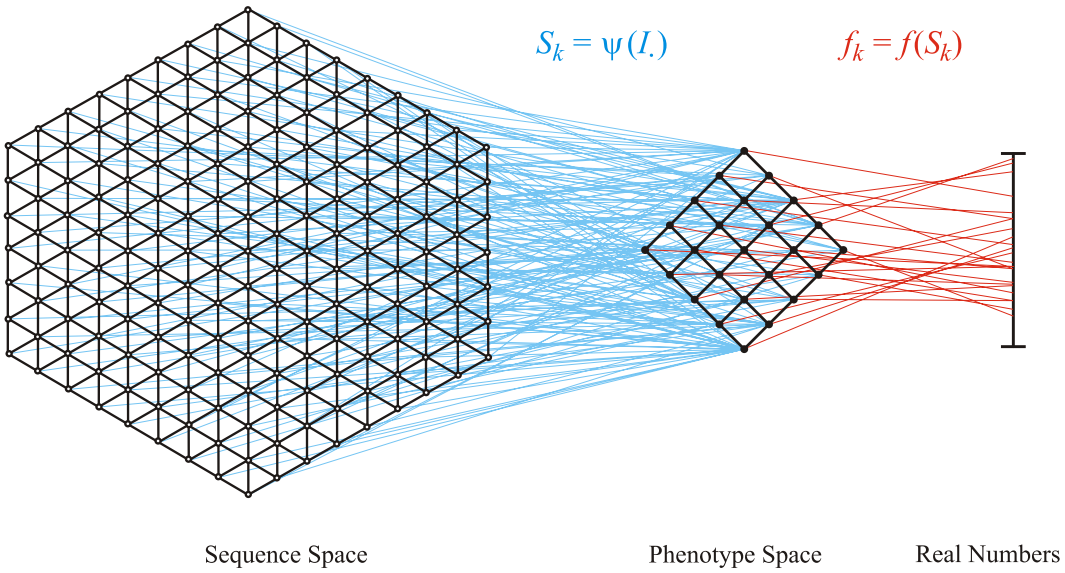



Figure 3.6: Mapping genotypes onto phenotypes and into fitness values.

to consider dynamics in phenotype space [40] the variables of all genotypes forming the same phenotype are lumped together. The relative frequency of phenotype $S_j$ is then expressed by $\eta_j = \sum_{i=n_{j-1}+1}^{n_j} \xi_i^{(j)}$.

The general relation of genotypes, phenotypes and fitness value is a two step mapping explained in section 1.2 and illustrated in figure 3.6. We consider the set of sequences folding into a given mfe structure $S_k$: $G_k = \psi^{-1}(S_k) \doteq \{I_j | \psi(I_j) = S_k\}$ in order to characterize the phenotypes by means of their preimages in sequence space. The set $G_k$ is transformed into a graph $\mathcal{G}_k$, which is called the neutral network of $S_k$. The edges connect all pairs of nodes with Hamming distance $d_{jk}^h = 1$. The Hamming distance [22] is a natural metric between sequences, while evolutionary relevant neighbourhood relations between phenotypes are much more difficult to derive [15, 16]. Topological details of phenotype space are described elsewhere [2, 3, 50], while we shall use only the frequency of occurrence of $S_j$ in the one-error neighbourhood of $S_k$, $\varrho(S_j; S_k)$, as defined in [16]:

$$\varrho(S_j; S_k) \;=\; \frac{\gamma_{jk}}{\ell \cdot (\kappa - 1) \cdot |G_j|} \tag{3.13}$$

with $\gamma_{jk}$ being the number of Hamming distance one contacts between the two neutral networks $\mathcal{G}_k$ and $\mathcal{G}_j$. The frequency of occurrence is not symmetric, because although the number of contacts is symmetric $\gamma_{jk} = \gamma_{kj}$, the size of the networks is different. Transitions matrices between phenotypes $\Omega = \{\Omega_{ij}; i, j = 1, \ldots, n\}$ are not symmetric in general $\Omega_{jk} \neq \Omega_{kj}$. Curve $a$ of figure 1.3 in the introduction show a rank ordered distribution of frequencies of occurrence. It may be separated into two regimes with different scaling properties: (i) A high frequency region which contains the frequent neighbours forming the statistical neighbourhood of the reference structure [16] and (ii) a low-frequency tail which fulfil a power-law that is analogous to "Zipf's law" [56].

With the frequency of occurrence closely related families of phenotypes can be defined:

$$\Upsilon_\varepsilon(S_k) \;=\; \{S_j \in \Sigma(S_k) | \varrho(S_j; S_k) \geq \varepsilon\} \;. \tag{3.14}$$

$0 < \varepsilon \leq 1$ denotes a minimum frequency of occurrence which define $\varepsilon$-neighbourhoods $\Upsilon$ of phenotypes $S_k$. Within these families of phenotypes $S^{(j)} = \{S_1^{(j)}, \ldots, S_k^{(j)}, \ldots, S_m^{(j)}\}$ transitions occur with a high probability and belong to the class of continuous transitions.

In order to investigate transition dynamics between phenotypes we compute first the probability of a mutation from phenotype $S_j$ to phenotype $S_k$:

$$\text{Prob}\{S_j \to S_k\} \;=\; \frac{M_j}{N} \cdot \ell \cdot p \cdot (1-p)^{\ell-1} \cdot \varrho(S_k; S_j) \;=\; \eta_j \, \Omega_{jk} \tag{3.15}$$

There is also a non-zero probability to obtain the phenotype $S_k$ by mutation of genotypes belonging to the neutral network $\mathcal{G}_k$, which is added to the probability of correct replication:

$$\text{Prob}\{S_k \to S_k\} \;=\; \frac{M_k}{N} \cdot \Big((1-p)^\ell \;+\; \ell \cdot p \cdot (1-p)^{\ell-1} \cdot \bar{\lambda}_k\Big) \;=\; \eta_k \, \Omega_{kk}$$

The restriction of low mutation rates allows us to simplify the expressions by approximating $(1-p)^\ell \approx 1 - \ell \cdot p$ and $\ell \cdot p \cdot (1-p)^{\ell-1} \approx \ell \cdot p$.

Next we extend replication-mutation dynamics to the level of families of phenotypes and focus on the phenotype $S_k^{(j)}$ at an instant when it is not (yet) present in the population. Then it is exclusively formed through single point mutations from all other members of the family $\mathbf{S}^{(j)}$ and thus we find:

$$\text{Prob}\{S_{i\neq k}^{(j)} \to S_k^{(j)}\} \;=\; \sum_{i=1,i\neq k}^{m} \eta_i\,\Omega_{ik} \;=\; l\cdot p\cdot \bar{\varrho}\sum_{i=1,i\neq k}^{m}\eta_i\;, \tag{3.16}$$

where we made use of the fact that the distribution of frequencies of occurrence suggests that the probabilities of continuous transitions between the members of family depend only on the relative frequencies $\eta_i$. In other words, replacement of individual $\varrho$-values by their mean,

$$\bar{\varrho} = \sum_{i=1}^{m}\sum_{j=1,j\neq i}^{m} \varrho(S_j;S_i)/\big(m(m-1)\big), \tag{3.17}$$

is assumed to be a sufficiently good approximation.

### 3.4.2 A Birth-and-Death Model

A birth-and-death model has to be kept as simple as possible to derive analytical expressions for population side effects [48]. $S^{(j)}$ denotes a family of phenotypes with equal fitness, whose members are accessible through continuous transitions. We shall focus on the single phenotype $S_k^{(j)}$, which is connected to other phenotypes $S_i^{(j)}$ of his family according to figure 3.7

The number of copies of phenotype $S_k^{(j)}$ is counted by the stochastic variable $\mathcal{X}$ with the probability distribution $\text{Prob}\{\mathcal{X}=X\} = P_X(t)$. All phenotypes of family $S^{(j)}$ except $S_k^{(j)}$ are lumped together and their total number is described by a stochastic variable $\mathcal{Y} = \sum_{i=1,i\neq k}^{m} M_i^{(j)}$. Since the main fraction of phenotypes belong to the family $S^{(j)}$ we neglect all others and set $\mathcal{X}+\mathcal{Y} = N_{\text{eff}} = N$ for convenience, where $N_{\text{eff}}$ is an effective population size. The time dependence is then modeled by the master equation of a linear birth-and-death process

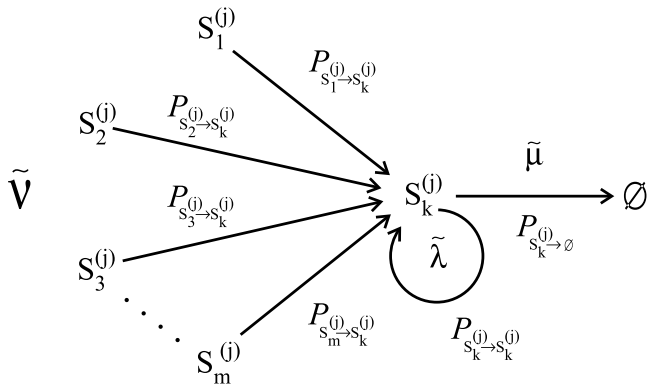



Figure 3.7: Transition probabilities between phenotypes $S_i^{(j)}$ and a particular phenotype $S_k^{(j)}$ which altogether belong to a family of phenotypes $S^{(j)}$ with equal fitness.

$$\frac{dP_X(t)}{dt} \;=\; \lambda_{X-1}\,P_{X-1}(t) \;-\; (\lambda_X + \mu_X)\;P_X(t) \;+\; \mu_{X+1}\,P_{X+1}(t)\;,$$

where the stochastic events of replication and mutation as measured by the transition probabilities $\tilde{\lambda}$ and $\tilde{\nu}$, respectively, are combined to yield

$$\lambda_X \;=\; \lambda \cdot X + \nu \;=\; \tilde{\lambda} \cdot X + \tilde{\nu}\,(N - X) \quad \text{with} \quad \lambda = \tilde{\lambda} - \tilde{\nu} \text{ and } \nu = N\tilde{\nu}\;.$$

The outflow of copies of $S_k^{(j)}$ from the reactor is described by $\mu_X = \mu \cdot X$ with $\mu = \tilde{\mu}$. From replication, mutation, and outflow probabilities discussed in the last subsection we obtain:

$$\lambda \;=\; \frac{1 \;-\; \ell \cdot p \cdot (1 - \bar{\lambda} - \bar{\varrho})}{N}\;, \quad \mu \;=\; \frac{1}{N}\;, \quad \text{and} \quad \nu \;=\; \ell \cdot p \cdot \bar{\varrho}\;. \qquad (3.18)$$

The parameters for birth ($\lambda$) and death ($\mu$) show inverse dependence on population size, whereas the immigration parameter ($\nu$) is independent.

### 3.4.3 Continuous Transitions

A qualitative description of this stochastic process is easy to derive: Since $\mu > \lambda$ is true, $\mathcal{X}(t)$ is predominantly decreasing without immigration. In reactors

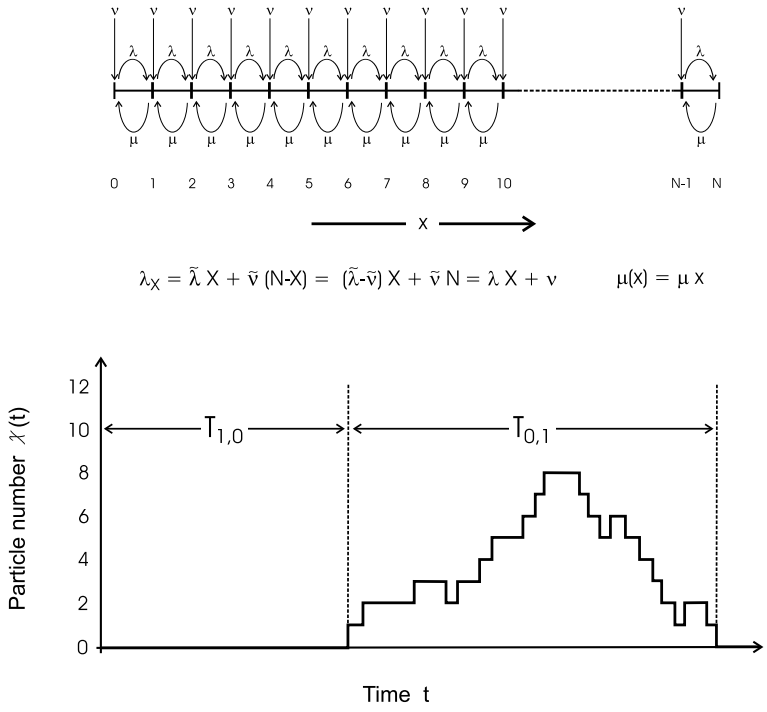



Figure 3.8: Birth-and-death process and first passage times.

with small population sizes $\nu$ is small compared to $\lambda$ and $\mu$. Whenever $S_k^{(j)}$ is formed by mutation from one of the other phenotypes $S_i^{(j)}$, $i \neq k$ it will be soon eliminated and therefore the number of continuous transitions is expected to be large in a relay series. On the other hand in large populations immigration is more dominant. A non-vanishing fraction of $S_k^{(j)}$ phenotypes will be sustained by sufficiently large mutation terms and much fewer continuous transitions can be observed in a relay series.

The process is described quantitatively with the assumption that the state $\mathcal{X} = N$ is a reflecting barrier. Since $\mu(N) > 0$ and $\lambda(N) > 0$ we need an approximation and set $\lambda(N) = 0$ and $P_X(t) = 0$ for $X \geq N + 1$. At the state $\mathcal{X} = 0$ we are concerned with a reflecting barrier because $\mu(0) = 0$ and $\lambda(0) < 0$. Since we are only interested in transitions between $\mathcal{X} = 0$ and $\mathcal{X} = 1$ the mentioned approximation will not strongly influence the results.

[20][2] describes stationary solutions as well as moments for first passage

[2]We remark that tables 2.3 and 2.4 contain serious errors: (i) A factor 'j' is missing in the

times for restricted birth-and-death processes between two reflecting barriers.

The expectation value for the first passage time to extinction $< T_{0,1} >$ is readily calculated from the birth-and-death parameters [20]:

$$< T_{0,1} > = \frac{1}{\mu} \sum_{k=1}^{N} \frac{\Pi_{1,k-1}}{k} = \frac{1}{\mu} \sum_{k=1}^{N} \frac{(k-1+\hat{\nu})_{k-1}}{k!\ \hat{\mu}^{k-1}} , \tag{3.19}$$

where we used $\hat{\mu} = \mu/\lambda$, $\hat{\nu} = \nu/\lambda$, and

$$\Pi_{i,j} = \frac{\lambda_i \cdot \lambda_{i+1} \cdot \ldots \cdot \lambda_j}{\mu_i \cdot \mu_{i+1} \cdot \ldots \cdot \mu_j} .$$

An expression for the variance of the first passage time, var $[T_{0,1}]$, is also available,

$$\text{var}\,[T_{0,1}] = < T_{0,1} >^2 + 2 \sum_{k=2}^{N} < T_{k-1,k} >^2 , \tag{3.20}$$

and we note larger scatter than expected for a Poisson process.

### 3.4.4 Discontinuous Transitions

On a plateau of constant mean fitness it happens regularly, that a fitter mutant is found, which can not establish the new fitness level and dies out soon after its creation. In order to derive a quantitative expression for the probability of survival for an advantageous sequence, we make use of essentially the same model as the one applied to the survival of mutants produced by continuous transitions.

The different situation requires an adjustment of the rate parameters referring to birth and death, which correspond to replication and mutation in the computer simulations: (i) We assume that the majority of sequences in the reactor are on the same fitness level and are therefore selectively neutral. In order to sustain constant population size these sequences replicate with a birth

expressions of higher moments $M_{N,m}^{(j)}$ and (ii) the minus sign in $V_{N,m}$ in table 2.4, second column should read '+'. pp.24-27

rate which is adjusted to the death rate. (ii) Since we consider only rare mutants of higher fitness the stochastic event of mutation will not be considered with sufficiently large probability within the time span considered.

The number of mutant molecules is like in the previous sections denoted by the stochastic variable $\mathcal{X}$. Compared to the dominant type, the new variant has an increased replication rate of

$$\lambda(X) \;=\; \frac{1+\vartheta}{N}\Big(1 \;-\; l\cdot p\,(1-\bar{\lambda})\Big)\,X \;.$$

The term containing $\bar{\rho}$ in equation 3.18 does not contribute here, because mutations are singular events. Unter assumption of balanced birth and death rate we find for the death rate

$$\mu(X) \;=\; \frac{1}{N^2}\Big(N - X \;+\; (1+\vartheta)\,X\Big)\,X \;=\; \frac{1}{N}\left(X \;+\; \vartheta\,\frac{X^2}{N}\right) \;.$$

Although we are dealing with a hart to solve non-linear birth and death process, analytical expressions are readily obtained for the two limiting cases with $\mu = 1/N$ and $\mu = (1+\vartheta)/N$.

If a single copy of a mutant is present in the population at time $t = 0$ its probability to suvive until at least time $t$ is given by

$$\sum_{i=1}^{N} P_{i,1}(t) \;=\; 1 \;-\; P_{0,1}(t) \;.$$

From the solution of the unrestricted process ($N = \infty$) it is possible to exactly compute the transition probability until the time when the maximal possible value of $\mathcal{X}$ reaches $N$ ($t > N$):

$$P_{0,1}(t) \;=\; \frac{1 \;-\; \exp\big((\lambda-\mu)\,t\big)}{1 \;-\; \frac{\lambda}{\mu}\exp\big((\lambda-\mu)\,t\big)} \;. \tag{3.21}$$

## 3.5 Survival Probabilities of New Structures

Through discontinuous transitions new structures are created, that are rarely reachable for the current population. They give an evolutionary optimization

| No | Structure | c/d | dist. |
|---|---|---|---|
| 1 | (((((....((((.......)))).)))))........((((((....((((.....)))))..))))))....  | c | 15 |
| 2 | (((((....((((.......)))).)))))........((((((...((.((.......)).)).))))))....  | d | 15 |
| 3 | ((((((...((((.......))))))))))........((((((...((.((.......)).)).))))))....  | c | 15 |
| 4 | ((((((...((((.......))))))))))........((((((...(((((.....)))))..))))))....  | c | 13 |

Figure 3.9: A detail of the continuous (c) and discontinuous (d) transitions in a relay series. The fitness neutral, discontinuous transition $1 \rightarrow 2$ opens 5 base pairs, while 4 of them are formed again, shifted by one base. (dist. shows the Hamming distance to the target structure.) After a continuous closing ($2 \rightarrow 3$) of a single base pair, the fitness innovation is made ($3 \rightarrow 4$). What is the survival probability of structure 2 ?

process the ability to establish new structural domains, which are itself or have the potential to reach fitter items. Usually discontinuous transitions are tantamount with fitness innovations, but also equal or lower fitness values, than the actual fittest sequences are possible. Especially if two consecutive discontinuous transitions can be observed, structures with equal or lower fitness than their ancestors can be found in the relay series. E.g. a shift, a roll over or a flip can be performed in two steps: The first step is a mutation to open a stack. Some of the released bases find other bases to pair before another mutation closes the shifted stack again. Both of these steps can be discontinuous transitions.

To examine the survival probability of such a new structure with equal fitness numerically a series of flow reactor runs with a average population size $N_{set} = 1\,000, 2\,000, 3\,000, 10\,000$ was started. The start population contains two different sequences, whose mfe-structures have Hamming distance $d^s_{0,\tau} = 2$ to the target shape. The population sizes are 1 and $N_{set} - 1$, respectively. Both structures require a major structural change to reach the target, which is very unlikely to occur within the limited number of replications of this experiment. (For illustration the structures of the start population and the target shape of runs with $N_{set} = 1000$ are shown.)

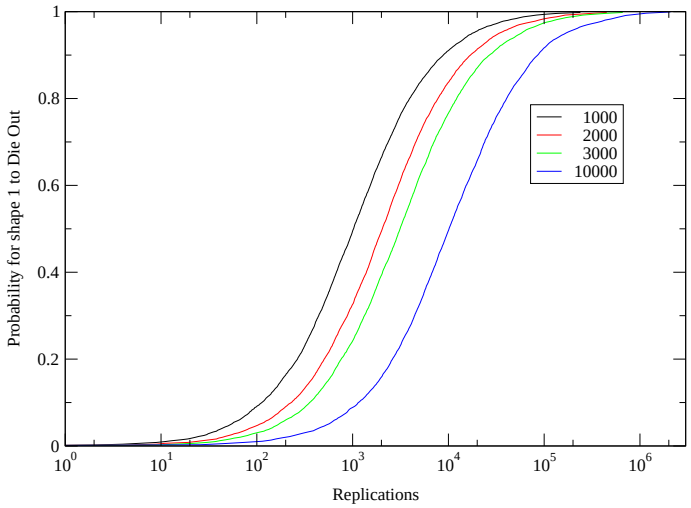



Figure 3.10: The die out probability of shape 1 represented by a single sequence in the start population vs. the number of replications. The different lines show the population sizes 1000, 2000, 3000 and 10000, respectively. This figure is based on 10 000 flow reactor runs for every population size and a mutation rate of 0.001 per sequence and base.

```
shape 1 ((((((...((((........))))..(((((......)))))).....(((((.......)))))..)))))).....   1
shape 2 ((((((...((((........)))).(((((.......)))))).....(((((......)))))..)))))).... 999
target  ((((((...((((........)))).(((((.......)))))).....(((((.......)))))..)))))).....
```

The mutation rate per base was 0 and 0.001, respectively. The flow reactor run was terminating after shape 1 died out. Even in the reactor runs with mutations, a transition to shape 1 never occurred, as well as the target structure was never found.

The probability $p$ to die out of a structure represented by a single sequence in a flow reactor with a low mutation rate ($m \leq 0.001$) is dependent on the number of replications $n_{rep}$ and the population size $N_{set}$ is almost perfectly described by the following function:

$$p(n_{rep}) \;=\; \left(1 + \frac{N_{set}}{n_{rep}}\right)^{-1} \tag{3.22}$$

In the limit of maximal mutation rate, every replication leads to a random

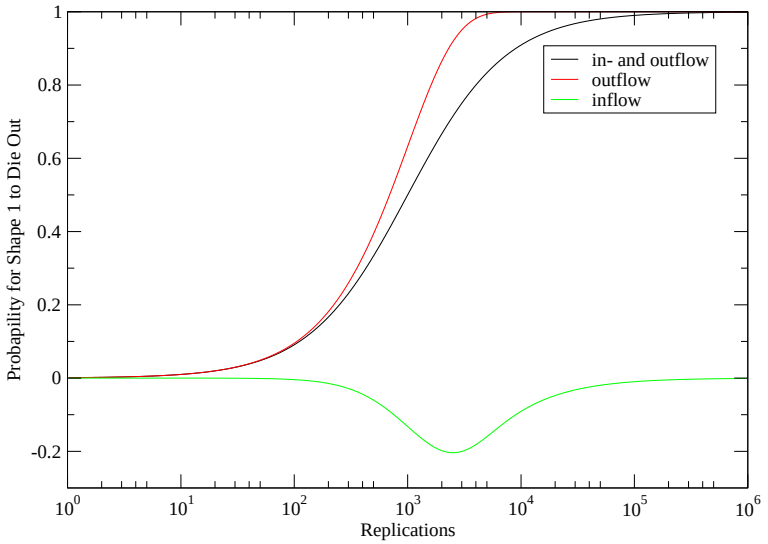



Figure 3.11: The die out probability of shape 1 at population size $N_set = 1000$ (black line) can be separated into two parts. If only outflow is possible (red line) the shape 1 dies out faster. Inflow though replication of the sequences of shape 1 diminish the die out probability (green line).

sequence and thereby also to a random shape. Under this conditions no amplification of shape 1 can be determined. According to our flow reactor model on every replication $n_{rep}$ one outflow reaction $n_{out}$ occurs on average.

$$n_{events} = 2\, n_{rep} = n_{rep} + n_{out} \tag{3.23}$$

The die out probability $p_{out}(n_{rep})$ for shape 1 if only outflow is possible is given by:

$$p_{out}(n_{out}) = 1 - \left( \frac{N_{set} - 1}{N_{set}} \right)^{n_{out}} \tag{3.24}$$

$$p_{in}(n_{rep}) = p(n_{rep}) - p_{out}(n_{out}) \tag{3.25}$$

$$p_{in}(n_{rep}) = \left(1 + \frac{N_{set}}{n_{rep}}\right)^{-1} + \left(\frac{N_{set} - 1}{N_{set}}\right)^{n_{rep}} - 1 \tag{3.26}$$

The difference $p_{in}(n_{rep}) \leq 0$ is the decrease of the die out probability in reactors with low mutation rate and denotes the amplification of the shape 1, which is due to replication and inflow from other structures.

## 3.6 Serial Transfer Experiment vs. Flow Reactor

In an *in vitro* serial transfer experiment a portion of the population is transferred to fresh stock solution, when a certain limit is reached. This can be a time limit, a certain population size, or other chemical or metabolic parameters (e.g. oxygen concentration, etc.). In our simulated flow reactors no stock solution or raw material is needed. Therefore the population can be simply diminished, when a given population size is reached, without loosing the relevant effects. The performance in terms of replications to find a target structure of serial transfer experiments in comparison to flow reactors in an *in silico* experiment was always unclear.

To achieve statistical significant results, a high number of runs under different conditions has to be compared. Because of the limited computer capacity and long folding times for longer RNA molecules, a rather short sequence is required for this experiment. Our target shape in all cases is a hairpin with 30 positions, a very common structure for this sequence length. The start sequence's mfe structure contains a bulge and requires a flip and some base pair closings to reach the target.

| | |
|---|---|
| start sequence: | `CAAGACUUUCCUCCACAGUUCGGAAUUUUG` |
| mfe-shape of start sequence: | `(((((..((((.........)))))))))` |
| target structure: | `((((((((((((....))))))))))))` |

The relevant criteria for the description of a serial transfer experiment are the number of sequences before and after the serial transfer, which will be

called the reduction step. The labels give these values in 1 000 sequence units. E.g. the setting 0.1/5 diminish the population to 100 sequences and let it grow up to 5 000 sequences before the next reduction. For comparison to the serial transfer experiments we take a flow reactor with average population size 1 000. The start population contains $N$ copies of the given start sequence, where $N$ is the population size immediately after a reduction step. The mutation rate is set to $p = 0.001$ per replication and base position. In 15 different experiments 1 000 runs each were performed.

The range of variation of such series of single runs is very high. The number of replications to reach the target in experiments using the same settings can vary by more than 2 orders of magnitude. The distribution is skewed and therefore the log-normal distribution was chosen. For convenience the values where $log_e$ transformed and treated as normal distributed. These results are shown in table 3.12.

To test for a significant difference between the performance of all settings an ANalysis Of VAriance (ANOVA) was made. The probability, that the number of replications to reach the target shape in different serial transfer reactors varies by chance is almost zero. Among the $n = 15$ different settings also all $(n^2 - n)/2 = 105$ pairwise comparisons were made. These results are shown in table 3.5. A more insightful sketch, which shows a connection between not distinguishable settings can be found in figure 3.13.

The runs show significantly better results for continuous reactors (label cont.) and reactors with a small reduction step (label 0.9/1.1). A high maximal population size leads also to significantly more replications to find the target shape. The reactor with 100 fold reduction and a maximal population size of 10 000 performs worse than any other settings in this experiment.

In contrast to flow reactors there is no development of distinct clusters because during expansion phase there is no fitness pressure which removes the sequences without neighbours. During the reduction phase sequences are removed by chance, which dilute the population density in the surrounding region of the start population. After some reduction steps without fitness increase the population is more and more equally distributed in sequence space

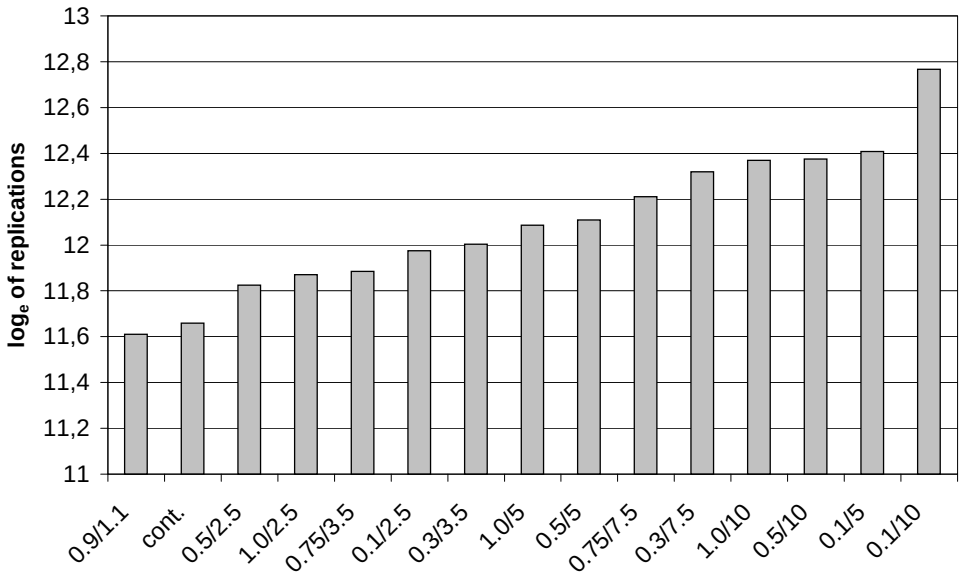



Figure 3.12: The arithmetic mean of $\log_e$ transformed replications to reach the target shape of 1 000 runs each of a flow reactor (label: cont) and 14 serial transfer experiments. The continuous reactor and the very similar 0.9/1.1 serial transfer experiment show the best performance. A small reduction during the serial transfer and a low maximal population size performs significantly better. Among the others experiments with a smaller maximal population size are mostly rated better than their counterpart with 5 000 to 10 000 sequences at maximal load.

Table 3.4: Serial transfer experiment performance. The table shows the arithmetic mean and variance of the $log_e$ transformed number of replications. The reduction is the quotient of maximal and minimal population size. A high reduction and high maximal population size leads to more replications to reach the target shape.

| Experiments Type | Reduction | Mean of $log_e(n_{rep})$ | Variance |
|---|---|---|---|
| 0.9/1.1 | 1.2 | 11,61 | 1,80 |
| cont. | - | 11,66 | 1,87 |
| 0.5/2.5 | 5.0 | 11,82 | 1,74 |
| 1.0/2.5 | 2.5 | 11,87 | 1,44 |
| 0.75/3.5 | 4,7 | 11,89 | 1,38 |
| 0.1/2.5 | 25.0 | 11,97 | 2,35 |
| 0.3/3.5 | 11.7 | 12,00 | 1,72 |
| 1.0/5 | 5.0 | 12,09 | 1,29 |
| 0.5/5 | 10.0 | 12,11 | 1,62 |
| 0.75/7.5 | 10.0 | 12,21 | 1,40 |
| 0.3/7.5 | 25.0 | 12,32 | 1,67 |
| 1.0/10 | 10.0 | 12,37 | 1,24 |
| 0.5/10 | 20.0 | 12,38 | 1,52 |
| 0.1/5 | 50.0 | 12,41 | 2,55 |
| 0.1/10 | 100.0 | 12,77 | 2,34 |

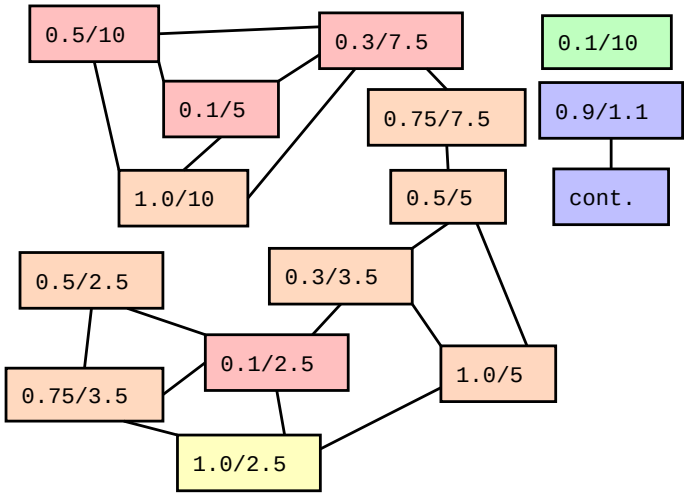



Figure 3.13: The performance in terms of replications to reach the target shape of the connected serial transfer reactor settings (vertices) can not be distinguished from chance. The labels give the population size in thousands sequences immediately after and before the serial transfer, respectively. The colors denote the reduction step (blue = 0...2, yellow = 2.5, orange = 4...12, red = 20...50, green = 100). The continuous (label cont.) and the nearly continuous reactor (label 0.9/1.1) performs better, the reactor with the highest reduction step (label 0.1/10) needs the most replications on average.

Table 3.5: The detection of significant performance differences between the serial transfer reactors and the flow reactor with an error rate of 0.05 are marked. A low maximal population size and a low reduction step performs better.

| | 0.9/1.1 | cont. | 0.5/2.5 | 1.0/2.5 | 0.75/3.5 | 0.1/2.5 | 0.3/3.5 | 1.0/5 | 0.5/5 | 0.75/7.5 | 0.3/7.5 | 1.0/10 | 0.5/10 | 0.5/10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| cont. | | | x | x | x | x | x | x | x | x | x | x | x | x |
| 0.5/2.5 | x | x | | | | x | x | x | x | x | x | x | x | x |
| 1.0/2.5 | x | x | | | | | x | x | x | x | x | x | x | x |
| 0.75/3.5 | x | x | | | | | x | x | x | x | x | x | x | x |
| 0.1/2.5 | x | x | x | | | | | | x | x | x | x | x | x |
| 0.3/3.5 | x | x | x | x | x | | | | | x | x | x | x | x |
| 1.0/5 | x | x | x | x | x | | | | | x | x | x | x | x |
| 0.5/5 | x | x | x | x | x | x | | | | | x | x | x | x |
| 0.75/7.5 | x | x | x | x | x | x | x | x | | | | x | x | x |
| 0.3/7.5 | x | x | x | x | x | x | x | x | x | | | | | |
| 1.0/10 | x | x | x | x | x | x | x | x | x | x | | | | |
| 0.5/10 | x | x | x | x | x | x | x | x | x | x | | | | |
| 0.1/5 | x | x | x | x | x | x | x | x | x | x | | | | |
| 0.1/10 | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

and it takes longer to find the possibility for a major transition. We can only speculate about the reasons for the lower performance: (i) The share of sequences, which have the highest currently assigned fitness seems to be lower, because unfit items are not removed immediately due to lack of fitness pressure. It is more likely to find a fitter item in the neighbourhood of a fit one than through the random-like migration in a serial transfer experiment. (ii) Due to the wider distribution in sequence space the formation of clusters is diminished or impossible. If no such neighbourhoods of neutral nets exists, which increase each other's population size by continuous transitions, fit shapes die out quicker and the chance to reach fitter structures is diminished. On the other hand there seem to be an advantage for serial transfer experiments: The problem of shifted structures described above (in section 3.3) can be bypassed by a two or more step transition without the fitness pressure of a flow reactor which rapidly removes unfit sequences. Irrespective of these assumptions there should be no difference between serial transfer experiments and flow reactors

if sequence space is homogeneous in terms of equal distribution of frequent structures. It is unlikely but possible, that the picture changes with different start and target structures.

## 3.7 The Phenotypic Error Threshold

The phenotypic error threshold [40] is the error rate in a flow reactor simulation, where an established fitness-superior shape is eliminated. Above this error rate almost any replication of a sequence leads to mutations as well as to a different mfe structure than their parent shape. Since one of the generic properties of folding is the shape space covering (in a radius of less than 15 mutations on a sequence with 100 nucleotides any frequent structure can be found. See also section 1.6) many mutations lead to a random shape. Below the error threshold a certain share of sequences fold into one of the shapes with maximal fitness at that time. In this section it is examined numerically, if the share of a master shape $\eta_m$, which is dependent on the error rate $p$, is also dependent on the average population size $N_{set}$.

$$\eta_m = f(p, N_{set}) \tag{3.27}$$

The start population of the test runs consists of $N_{set}$ copies of a sequence, whose mfe-structure is the target shape. In such a population no fitness increase is possible through a major transition, which would shift the ratios of fit and unfit shapes and bias the results.

```
start sequence ACGCGUAUCGGGCAUAGCGUCGCCCAGGCGAAAAUUACUCGCCCAGAACUUACCGACAUCGUAGGGGCGCGUCUAC
       target ((((((...((((........)))).(((((.......))))).....(((((.......))))).)))))).....
```

Computations of all combinations of the error rate $p$ between 0.003 and 0.036 in 0.003 steps and the population sizes $N_{set}$ = 1 000, 2 000, 3 000, and 10 000 were performed. The fitness for every sequence was computed using equation 3.1, after calculating its mfe-structure and its Hamming distance to the target shape.

Table 3.6: The share of the master shape $\eta_m$ at different error rates $p$, which is independent on the population size $N_{set}$.

| | Population size $N_{set}$ | | | |
|---|---|---|---|---|
| $p$ | 1000 | 2000 | 3000 | 10000 |
| 0.003 | 0.857 | 0.861 | 0.859 | 0.860 |
| 0.006 | 0.735 | 0.737 | 0.737 | 0.736 |
| 0.009 | 0.633 | 0.632 | 0.632 | 0.633 |
| 0.012 | 0.537 | 0.542 | 0.541 | 0.540 |
| 0.015 | 0.461 | 0.461 | 0.466 | 0.465 |
| 0.018 | 0.398 | 0.394 | 0.399 | 0.397 |
| 0.021 | 0.336 | 0.340 | 0.338 | 0.338 |
| 0.024 | 0.280 | 0.285 | 0.288 | 0.288 |
| 0.027 | 0.240 | 0.243 | 0.242 | 0.243 |
| 0.030 | 0.205 | 0.206 | 0.206 | 0.207 |
| 0.033 | 0.168 | 0.176 | 0.174 | 0.173 |
| 0.036 | 0.145 | 0.145 | 0.146 | 0.148 |

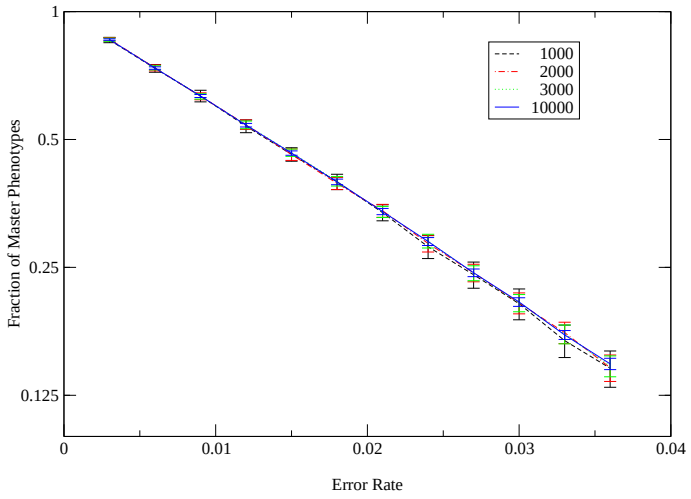



Figure 3.14: The linear/log plot of the error rate $p$ vs. share of master shapes in the total population $\eta_m$, shows the independence on the population size stated in the legend.

Until the termination of the flow reactor runs at time unit 10, every 0.1 time units, the share of sequence copies with target shape $\theta_t^{N_{set}}$ was examined. The results $\eta_m$ are the arithmetic mean of all $\theta_t^{N_{set}}$.

$$\eta_m^{N_{set}} = \frac{1}{n} \cdot \sum_{k=1}^{n} \theta_k^{N_{set}} \tag{3.28}$$

$\eta_m^{N_{set}}$ show no significant difference for the different population sizes. Therefore it can be stated, that the share of master-sequences is only dependent on the mutation rate but it is independent on the population size.

If shapes in the population have a minimal Hamming distance $d_{ij}^s > 0$ the share of fittest sequences is smaller because due to our fitness function also the fitness difference between fittest and unfitter structures is smaller.

## 3.8 A Lineage Sample Run

One the main research targets of the study of *in silico* evolution are the directly involved sequences and structures. Like in the preceding chapters the term *sequence* is used as a shorthand for a *RNA sequence type* with a defined primary structure of consecutive nucleotids, for which a certain number of *sequence copies* is present in the reactor. It is important to state that in the current simulation the fitness of a sequence is based on its mfe-structure and its distance to the target structure (see also equation 3.1).

The inheritance relation in a flow reactor simulation can be evaluated at different levels of accuracy:

(i) The *relay series*, which are described in detail in chapter 3.2, are based on structures and is the simplest form of a inheritance relation series. The creation and extinction time as well as the shape of origin (the mfe-structure of the replication's template-sequences) of every structure is recorded. (ii) The *genealogy* logs the lifespans of sequences, and tracks back the creation of sequences analogous to the relay series. (iii) The *lineage* records the production and outflow of every sequence **copy**. It is the most precise way to reflect the evolutionary process.

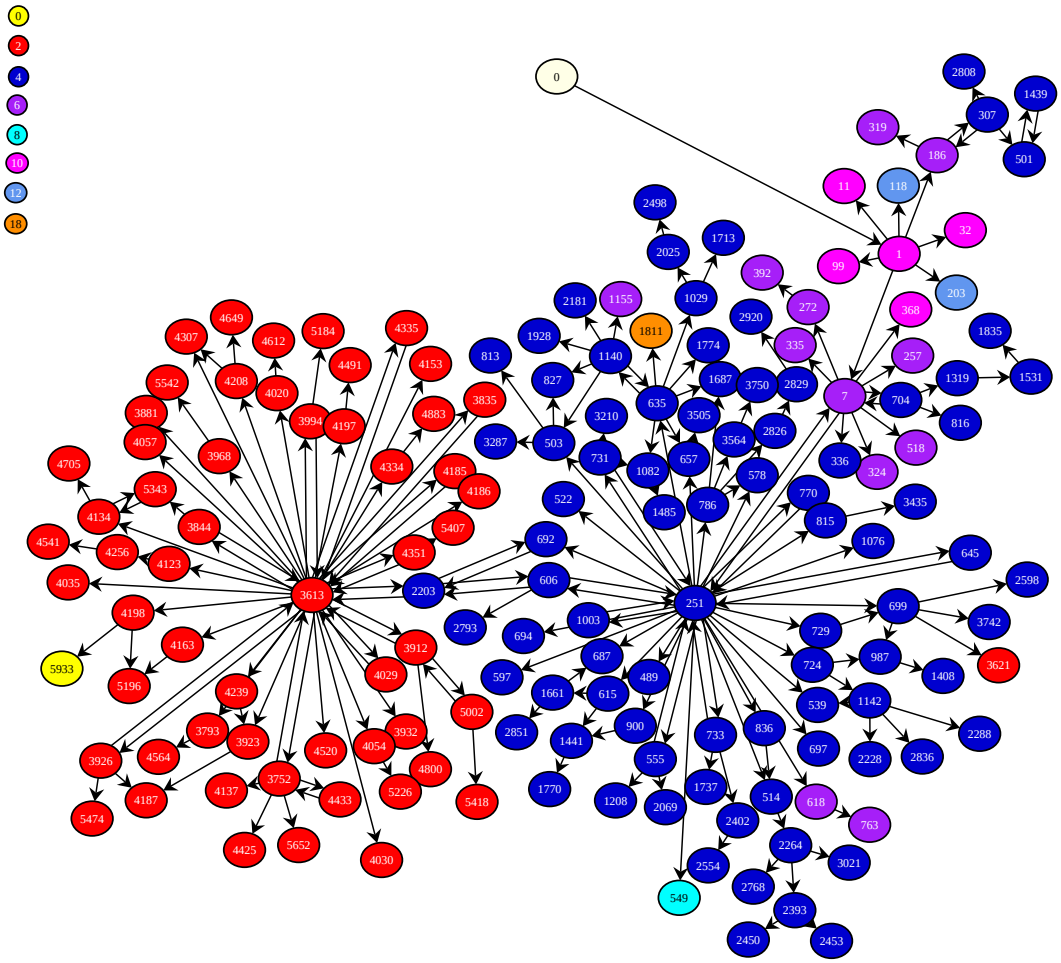



Figure 3.15: The lineage of a flow reactor run. The circles denote sequence types, while the arrows indicate the ancestor and descendant of a mutational replication. The colors show the Hamming distance of their mfe-structure to the target shape. See the legend in the left upper corner. Most sequences without influence are removed from the sketch in the following way: In 3 iterating step all sequences without (drawn in) descendant or with only its ancestor as a descendant are removed from the plot, whereas the yellow target sequence No. 5933 is preserved. Cycle 0 is the experimentor, who puts in sequence 1 as a start population. The lineage tracks the following path of sequences: $1 \rightarrow 7 \rightarrow 251 \rightarrow 606 \rightarrow 251 \rightarrow 692 \rightarrow 2203 \rightarrow 3613 \rightarrow 4198 \rightarrow 5933$

Table 3.7: The creation of more precise types of inheritance relation series require increasing computational effort.

| **Inheritance Relation Series** | **Based on** | **Accuracy** | **Computer Memory Requirements** |
|---|---|---|---|
| Relay Series | Structures | low | low |
| Genealogy | Sequences | middle | high |
| Lineage | Sequence copies | high | high |

To explain the relation of sequences and their role in the evolutionary optimization process, we monitor a short sample run.

```
                    1         2         3         4         5         6         7
          ruler     12345678901234567890123456789012345678901234567890123456789012345678901234567890123456
start sequence      UCGGAUGAAUGCAUGUCGGAGCGUAACUAGAACGAGAAUAUGGUGGAAUAAGGGACUGUACCUUAAAUCCGAUCUU
  start shape       ((((((..((((........)))).((((..........)))).....(((((.......)))))..)))))).....
 sequence 251       UCGGAUGAAUGCAUGUCGGAGUGUAACUAUAACGAGAAUAUGGUGGAAUAAGGGACUGUACCUUAAAUCCGAUCUU
 shape of 251       ((((((...(((((......)))))((((((.......))))))....(((((.......)))))..)))))).....
sequence 3613       UCGGAUAAAUGCAAGUCGGAGUGUAACUAUAACGGGAAUAUGGUGGAAUAAGGGACUGUACCUUAAAUCCGAUCUU
sequence 4198       UCGGAUAAAUGCAAGUCGGAGUGUAACUAUAGCGGGAAUAUGGUGGAAUAAGGGACUGUACCUUAAAUCCGAUCUU
3613 and 4198       ((((((...((((........))))((((((.......))))))....(((((.......)))))..)))))).....
sequence 5933       UCGGAUAAAUGCAAGUCGGAGUGUAACUAUAGCGGGAAUAUGGGGGAAUAAGGGACUGUACCUUAAAUCCGAUCUU
  target shape      ((((((...((((........)))).(((((.......)))))).....(((((.......)))))..)))))).....
```

About 150.000 replications are needed to find the 76 bases clover leaf target shape (see also figure 3.1) from a structure with Hamming distance 6. From the 5933 different sequences only 10 are involved in the lineage. 1312 different shapes are found in this run, while only 5 are part of the lineage.

Figure 3.15 shows the relation of sequences. From the start population, which are 1 000 sequence copies of type 1, sequence 7 is created after 0.0095 time units. The sequence type 7 is the ancestor of type 251, the most import sequence of the flow reactor run. The creation of the first copy of sequence 251 was both, a fitness increase and a discontinuous transition. Since this was a unique event, there are no other sources of sequence copies with such a high fitness at that time. With a replication accuracy of 0.999 per sequence and base, replications of sequence 251 leads mainly to new copies of the same type,

Table 3.8: The most important replications leading to mutations in the lineage sample run. A move on the neutral net (692 → 2203) turns out to be a bottleneck in the lineage.

| ancestor | descendant | frequency | comment |
|---|---|---|---|
| 1 | 7 | 5 | |
| 7 | 251 | 2 | second time after 606 and 692 appeared |
| 251 | 606 | 18 | |
| 251 | 692 | 9 | |
| 606 | 2203 | 1 | dies immediately (no descendants) |
| 606 | 251 | 1 | |
| 692 | 2203 | 1 | first bottleneck |
| 2203 | 3613 | 1 | second bottleneck and fitness jump |
| 2203 | 692 | 2 | |
| 2203 | 606 | 2 | |
| 3613 | 2203 | 8 | |
| 3613 | 4198 | 12 | |
| 4198 | 5933 | 1 | |

increasing its population share exponentially. Then other sequence types with the same fitness are found, slowing down and further limiting the enhanced reproduction of type 251. Nevertheless after a short time more than 50% of all sequences in the reactor were copies of type 251 (see Figure 3.16). The important role of this sequence type can be emphasized by the fact, that in 20.6% of all replications of this flow reactor run, sequence type 251 was the ancestor. It's clear that all $3\ell = 228$ of its one point mutants can be found among them. The opening of a single base pair between $U_{14}$ and $G_{21}$ or between $A_{26}$ and $U_{44}$ would decrease the distance to the target. But mutations on one of these positions either stabilize the shape ($C_{14}$ as well as $A_{21}$) or lead to completely different mfe-structures.

After sequence 251 the lineage goes over 3 sequences on the same neutral net. One of these sequences (2203) is created only once by its ancestor in the relay series. Although this is no shape transition, it is a bottleneck in the lineage. Every sequence created after the following fitness increase is a

descendant of this sequence copy. This is not uncommon because sequences on the same fitness level found after the major transition at most only get the chance to amplify itself and produce not more than a few neighbours, many of them on the same neutral net. If the sequence of such a mutational step or one of its descendants make a major transition, it appears as a bottleneck in the final backtracking procedure to create the *lineage*.

The next step in the lineage is a point mutation ($U_{14} \rightarrow A_{14}$) which shortens a stack by opening a wobble base pair (type 3613, ancestor in $19,2\%$ of all replications) and leads to a fitness increase. After a silent mutation (type 4198) the target is reached ($U_{44} \rightarrow G_{44}$) with sequence 5933, which supplant the former population in a few time units.

## 3.9 Comparison of Relay Series and Lineage

The *relay series* [15, 16, 45] (a detailed description can be found in section 3.2) give an insight into the relation of structures in a flow reactor run. From the structures of the start population to the target shape it can be understood now, how continuous and discontinuous transitions are involved into this evolutionary simulation.

From the target shape $\alpha$ the *relay series* goes back to the ancestor $\beta$ of $\alpha$ which made $\alpha$ a structure innovation, a new or newly recreated shape in the actual population. Then the structure innovation ancestor of $\beta$ is searched. In iterating steps one can trace back to a shape $\omega$, a structure in the start population. This backtracking is based on shapes and does not consider the sequences, that are the basis for these structures.

The sequences of the lineage can be easily transferred into structures. This inheritance relation will be called *structure lineage series* and can be easily compared with the *relay series*.

There are two important differences between the relay series and the structure lineage series: The *relay series* considers only structure innovations, the creation of a shape which is not part of the actual population and disregards all other structure productions. Only the creation and extinction of a structure is

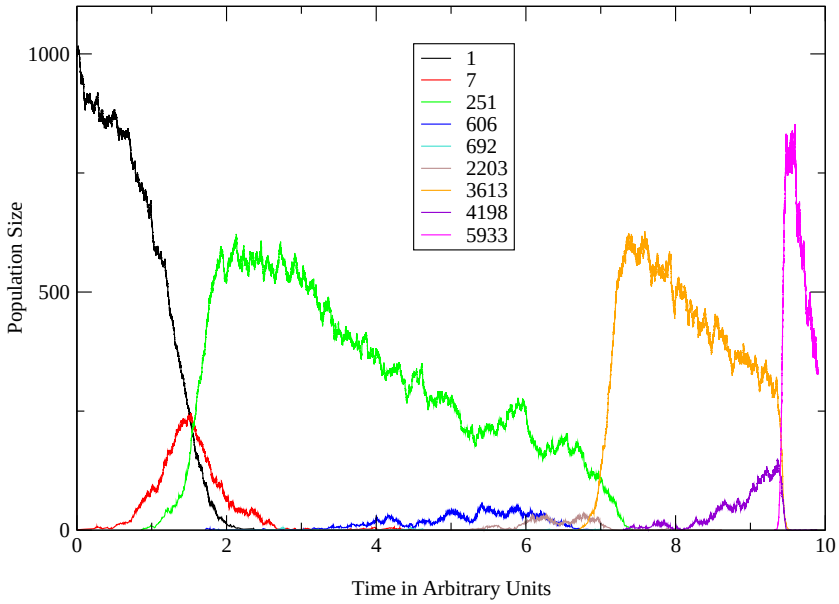



Figure 3.16: The lifespan and population size of selected sequences involved in the lineage. Being one of the first sequences with a higher fitness, than the actual population, increase the chance to play an important role. Sequence type 251 created 300 different sequence types, including all 228 sequences with distance 1. All neighbours where also created by sequence 3613, which was the first one with Hamming distance 2 and is also major player in this lineage. When other sequences on the same neutral net (like 4198) are established the population share of 3613 decreases. After the target is reached with sequence 5933, the former population dies out immediately.

recorded, regardless whether this shape is recreated again within the lifespan. (i) If the shape of origin of the recreation is another structure, the ancestry is always assigned to the first producer within a lifespan in the relay series. (ii) The template sequence in a recreation of a shape can be on a distant position in sequence space of their first ancestor sequence. The origin of this sequences can be a very different shape. In both cases the real origin is blurred.

For both inheritance series the finding time of the target structure has to be defined. Due to the stochastic outflow procedure sometimes the target structure dies out again. We set the time $t$ as the first appearance of the target shape, that never dies out again. For the lineage the underlying sequence copy $S(t)$ is the last one in the inheritance back trace to a sequence copy $S(0)$ in the start population.

In the 36 *relay series* steps 37 structures are involved, while the *structure lineage series* consists of 41 different structures in 49 steps. Of the total of 45 different structures 8 are unique in the *structure lineage series* and 4 can only be found in the *relay series*. The shape overlap is very high, which means that in general the same structures are involved in the series.

The comparison of the succession of shapes shows that the same discontinuous transitions can be found in the *relay series* and in the *structure lineage series*. Usually these transitions are the only connection between the population before and after this event. Therefore any structure inheritance series of this run passes that way. The continuous transitions between two discontinuous transitions follow another path and flips between structures of the same boundary than the shapes of the relay series.

## 3.10 Movement and Spreading in Sequence Space and Shape Space

Evolutionary optimizations in flow reactors proceed on two time scales. Fast periods containing cascades of adaptive changes are interrupted by long quasi-stationary epochs of neutral evolution, during which populations drift randomly on neutral networks, until a neighbourhood is found, where a mutation

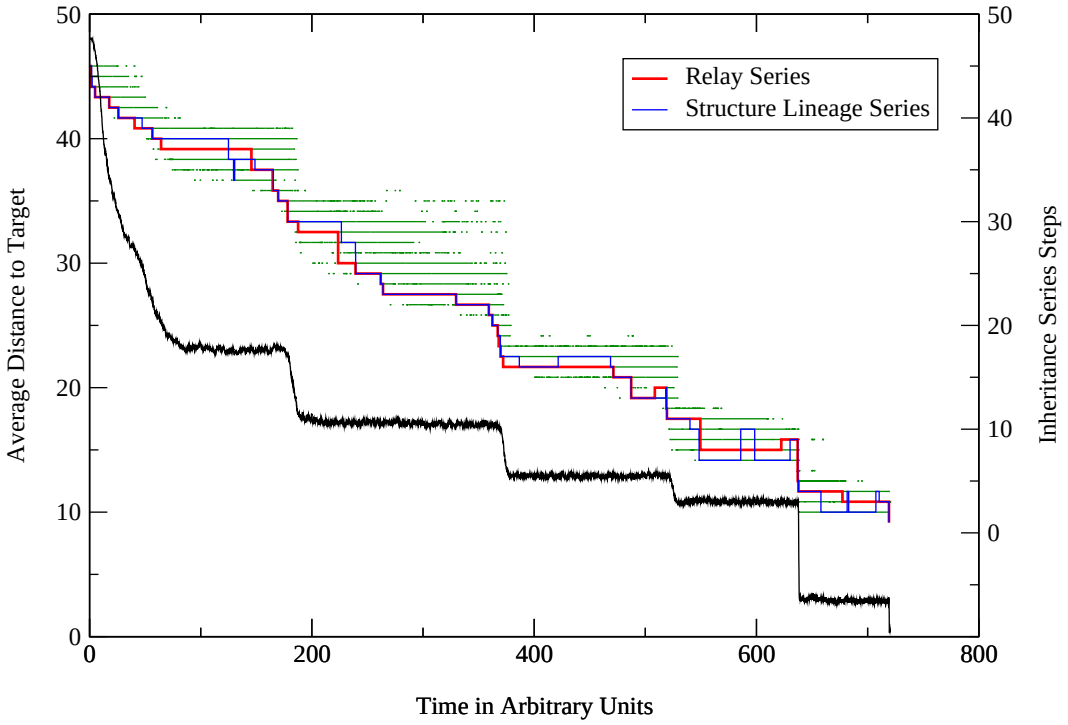



Figure 3.17: Comparison of the *relay series* and the *structure lineage series* in a flow reactor run with average population size $N_{set} = 2\,000$ and a mutation rate of $p = 0.001$ per base and replication. The sequence copies of the lineage were folded into their mfe-structure to allow a comparison with the relay series. The green lines show the lifespans of the structures involved in at least one of the series in chronological order of their first appearance. During discontinuous transitions the two series align with each other.

leads to the next major transition.

Similar observations where made in an elegant serial transfer experiment with *Escherichia coli*, which was carried out by Richard Lensky and coworkers [11,31,36]. For more than 3 years populations of $5 \times 10^8$ cells where diluted 1:100 every day, leading to about 10 000 generations of cells with a average generation time of 2.6 hours. The phenotypic evolution in terms of cell size was compared with the genotypic changes determined through DNA fingerprinting [36]. Only in the initial period, fast phenotypic changes happened. In contrast the genotypic changes took place, while little phenotypic evolution was determined in the saturation phase. In this section it is investigated, if this contrary phenomenon of genotypic/phenotypic evolution can also be observed in simulated flow reactor populations.

The usual time axis is replaced by replications $r$, in order to come as close as possible to the number of generations, where on average one generation corresponds to $N_{set}$ replications. To visualize the diversity of genotypes over time we apply to measures: (i) The first one is the mean Hamming distance within or between populations:

$$d_P(r, \Delta r) \;=\; \frac{\sum_{j=1}^{N(r)} \sum_{k=1}^{N(r+\Delta r)} d^{\,h}(I_j, I_k)}{N(r) \cdot N(r+\Delta r)} \tag{3.29}$$

It describes the spreading of the population in sequence space and is a appropriate measure of the diameter of the mutant cloud. The size of the mutant cloud increases with time on fitness plateaus and drops immediately at fitness increases, which happen usually concurrently with a discontinuous transitions at the end of an quasi-stationary epoch.

(ii) The other measure is the distance between the mean nucleotide sequences at two different times $t$ and $t + \Delta t$:

$$d_C(r, \Delta r) \;=\; \sum_{k=1}^{\ell} \sqrt{1 \;-\; \sum_{j=\mathbf{A},\mathbf{U},\mathbf{G},\mathbf{C}} \pi_j^{(k)}(r)\, \pi_j^{(r)}(r+\Delta r)} \tag{3.30}$$

The vector $\vec{\pi}^{(k)}(r) = \{\pi_{\mathbf{A}}^{(k)}, \pi_{\mathbf{U}}^{(k)}, \pi_{\mathbf{G}}^{(k)}, \pi_{\mathbf{C}}^{(k)}\}$ is the square-normalized distri-

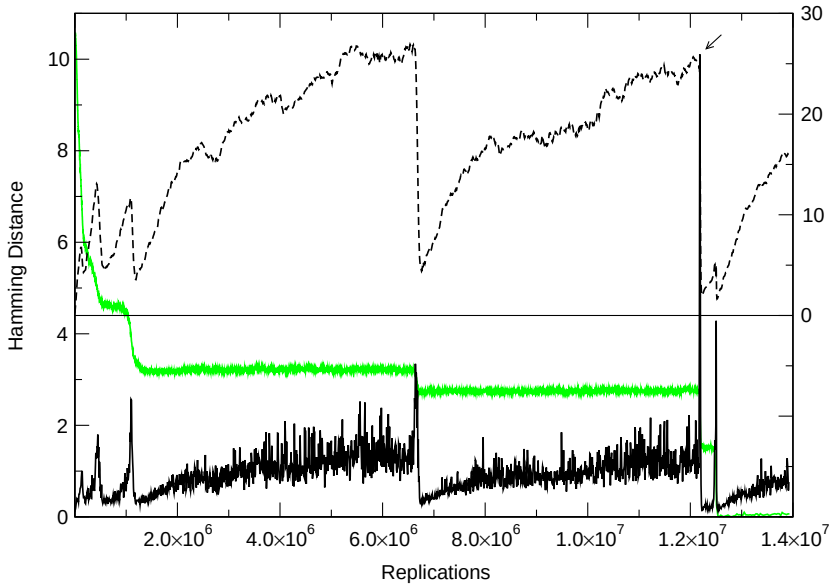



Figure 3.18: **Variability in genotype space during punctuated evolution**. Shown are the results of a simulation of RNA optimization towards a tRNA target with population size $n = 3000$ and mutation rate $p = 0.001$ per site and replication. The figure contains two plots with different measures of genetic diversity, $d_P(r, \Delta r)$ and $d_C(r, \Delta r)$ with $\Delta r = 8\,000$ replications, against time, which is expressed as the total number of replications performed so far, and the trace of the underlying trajectory in green recording average distance from target. The upper plot contains the mean Hamming distance between the population ($d_P$; dotted line, right ordinate) at time $t$ and time $r + \Delta r$ and the lower one shows the Hamming distance between the mean sequences at the same moments ($d_C$; full line, left ordinate). The arrow indicates a remarkably sharp peak of $d_C(r, 8\,000)$ at the end of the second long plateau which reaches a Hamming distance of about 10. Every adaptive phase is accompanied by a drastic reduction in the genetic diversity while genetic variation increases during quasi-stationary epochs. The mutant cloud, whose average size is expressed by $d_P(r, \Delta r)$, expands fast during neutral evolution and reaches diameters up to Hamming distance 25 whereas the center of the cloud migrates only at a speed of Hamming distance 1 per 8 000 replications.

bution of nucleotides at position $k$: $\pi_i^{(k)} = \alpha_i^{(k)} \;/\; \sqrt{\sum_{j=\mathbf{A},\mathbf{U},\mathbf{G},\mathbf{C}} \left(\alpha_j^{(k)}\right)^2}$ with $\sum_{j=\mathbf{A},\mathbf{U},\mathbf{G},\mathbf{C}} \alpha_j^{(k)} = 1$.

$d_C(r, \Delta r)$ is in essence a measure for the movement of the center of the distribution of sequences. It also increases during the quasi-stationary phase and saturates at values slightly above Hamming distance 1 per 8 000 replications. This is due to the increasing size of the mutant cloud (increasing $d_P(r, \Delta r)$), which leads to a quicker move of the center, if a distant cluster of similar sequences amplifies or dies out.

At the end of every epoch a sharp peak can be found, which indicates a quick relocate of the center of the mutant cloud. This peak is due to a major transition, which leads to a fitness increase. The whole population is supplanted quickly and the fitter sequence forms the new center of the mutant cloud. The position of the bottleneck sequences relative to the former mutant cloud specify the height of the peak. If it was located far away from the former center in terms of Hamming distance the peak turns out to be higher. But in any case it is limited by the maximal Hamming distance of any sequence pair in the actual reactor population.

Although the picture of genotypic versus phenotypic evolution obtained from *in silico* experiments is much more detailed than the results recorded with E. coli [36], we see an asynchronous speed in phenotypic and genotypic evolution. The genotypic evolution is faster during the phases of phenotypic stasis and vice versa.

## 3.11 Principal Components Analysis

From the former section we know, that after a fitness increase the sequences in a flow reactor simulation spread in sequence space. The diameter of the mutant cloud increases, while the phenotypic evolution is humble.

In this section we want to observe graphically, if the spreading in sequence space is uniformly distributed or the formation of clusters can be discovered.

For the analysis we take the population of the first fitness plateau of a flow

reactor run shown in figure 3.2 and 3.18 at 1.2, 1.264, 1.36, 2.8, 4.0, 5.2, 6.4, and $6.84 \times 10^6$ replications. To display the relation of the individual sequences in a 2D plot, the genotype distributions were transformed to the principal axes, and the individual sequences were projected by the 2 largest eigenvectors:

With the sequences with length $\ell$ of each of this reactor dumps the $4\ell \times 4\ell$ variance-covariance matrices were created. Every base of a sequence is split into 4 binary digits, where presence or absence of a base is encoded with 1 or 0, respectively ($A \to 0001$, $U \to 0010$, $G \to 0100$, $C \to 1000$)

Scaled with the eigenvalue the scalar product of the largest eigenvectors of the variance-covariance matrix and the converted RNA sequence give the x-axis position and the scaled scalar product with the second-largest eigenvector the y-axis position in the 2 dimensional plot. The number of copies of a sequence is pictured with the size of the dot, the distance to the target is color encoded.

To observe a continuous picture of the clustering of reactor populations, reactor dumps after every 8000 replications were treated with the procedure described above. It can be show that during the optimization procedure distinct clusters are created and the formation, disaggregation, and splitting of clusters can be observed, although due to the limitation to two dimensions this visualization method doesn't necessarily reflect real distances between all existing sequences.

## 3.12 Cluster Analysis

A major fitness transition in a flow reactor run is (except at the start) a very rare event, where a fitter sequence is found, that survives in the reactor. Its offsprings will soon replace the whole population, while the relatively unfit former sequences die out. Mutations during the replication produce sequences in the neighbourhood, that sometimes have the same fitness as their ancestors. Some of them are on the same neutral net, others find new, equally fit structures. Other unfit sequences usually have a limited life time in the reactor. After some replications the newly produced sequences spread in the neighbourhood. Are the sequence in this propagation process clustered to closely related

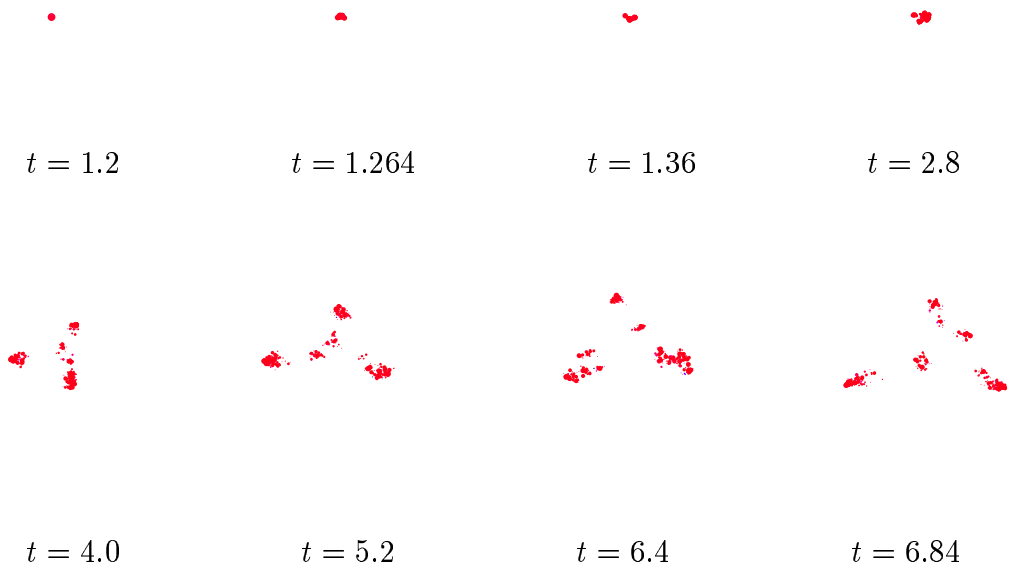



Figure 3.19: **Spreading of a population in genotype space during a quasi-stationary epoch**. The individual figures are snapshots of the genotype distribution at times corresponding to 1.2, 1.264, 1.36, 2.8, 4.0, 5.2, 6.4, and $6.84 \times 10^6$ replications. In order to visualize spreading, genotype distributions were transformed to principal axes and individual sequences were projected onto the plane spanned by the two largest eigenvectors. Along the series we observe an important and characteristic feature of population spreading in neutral evolution: The populations break up in smaller subclusters which diffuse radially away from the center of the distribution (See also the model on neutral evolution discussed in [4, 27]). Whenever an innovation with increase in fitness happens in one of the subclusters this subcluster takes over further development and all other subclusters die out.

groups or do they spread individually?

One approach of cluster assignment is Ward's minimum variance method [53]. At the beginnig every single sequence is treated as a separate cluster.

$$\bar{x}_k = \frac{1}{n_A}\sum_{i\in A} X_{k,i} \tag{3.31}$$

$$SAQ(X_k|A) = \sum_{i\in A}(X_{k,i}-\bar{x}_k)^2 \tag{3.32}$$

$$SAQ(X|A) = \sum_{k=1}^{m} SAQ(X_k|A) = \sum_{k=1}^{m}\sum_{i\in A}(X_{k,i}-\bar{x}_k)^2 \tag{3.33}$$

$$d(A,B) = SAQ(X|A\cup B) - SAQ(X|A) - SAQ(X|B) \tag{3.34}$$

In iterating steps clusters with minimal $d(A,B)$ are merged until one large cluster remains or a given $d(A,B)$ is exceeded. Although Ward's algorithm seem to be the method of choice for many clustering problems, it has two important drawbacks for RNA sequence populations. (i) It usually produces clusters of similar size and (ii) sometimes creates structure in uncorelated populations. If groups of RNA molecules with a distinct nucleotide sequence are formed in a flow reactor population, they probably could also die out without exchange to the remaining population. Ward's clustering method tend to merge a small uncorrelated cluster to a larger one, giving the impression that smaller clusters don't exist. Decrease in population size of a subcluster over time, down to dying out can therefore never be observed in a Ward cluster.

Another kind to find groups of correlated sequences, which can clearly be seperated from the rest of a flow reactor population, is to create a complete graph, whose vertices are the RNA sequences and whose edge weights are the Hamming distance between them. If all edges are removed, whose weight exceeds a given limit $d_{ij}^h > \phi$, the graph splits into clusters. This method will be called subgraph-clustering.

To examine the existance and formation of distinct clusters, a flow reactor with average population size 3000 and replication accuracy of 0.999 per base is started. The target shape is a tRNA clover leaf consisting of 76 bases (see figure 3.1). The mfe structure of the inital sequences is the target shape and therfore this run has no major fitness transition. Every 2000 replications a subgraph-clustering of the whole population with subcluster threshold Hamming distance $\phi = 9$ is performed. Around 73% of the sequences can be found in every

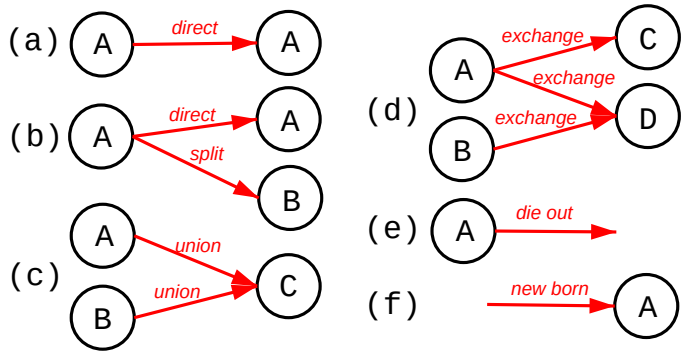



Figure 3.20: Cluster transition types: (a) shows the most common type of cluster transition. All sequences of cluster A at $r$ replications, that survived until $r + \Delta r$ replications can be found in one cluster, and vice versa. (b) sequences of cluster $A_r$ can be found in 2 or more clusters at $r + \Delta r$ replications. The transition to cluster $A_{r+\Delta r}$ contains the highest population number and is therfore the successor of the transition. (c) union of two clusters, no successor can be assigned. (d) exchange of sequences between clusters, also no successor can be determined. (e) no sequence of cluster A can be found in any cluster at $r + \Delta r$ replications. (f) the ancestors of the sequences of a new cluster cannot be found in any previous cluster

following dump. Then consecutive groups of clusters are compared and their transitions are classified according to figure 3.20.

97.23% of all clusters have a direct successor (type (a) and partly type(b)) in the next population dump. That implies the existance of distinct clusters that have mostly no sequence exchange with other clusters and validates the method of subgraph-clustering for this approach. This succession is called a cluster series. Because of the movement in sequence space clusters or at least some individuals of a cluster can come closer to each other exceeding the minimal Hamming distance beetween any member and combining two clusters. This union (c) is recorded for 2.28% of the clusters. In 0.33% they die out (e) and in 0.16% the exchange (d) between two clusters happend, respectively.

How do new clusters arise? 53.82% are splits from other clusters (type (b)), while 39.85% are products of exchange (type (d)) and 6.19% come into existence by union (type (c)). The remaining 0.39% have no direct ancestor

(type (f)), either because the ancestor sequences died out or the sequences of the clusters mutated so quickly between the dumps that no ancestor can be examined.

To observe the spreading in sequence space the mean Hamming distance within a population is given by

$$d_P(t) \;=\; \frac{\sum_{j=1}^{N(t)} \sum_{k=1}^{N(t)} d^h(I_j, I_k)}{(N(t))^2}$$

The highest average Hamming distance within a population of this flow reactor run is around 23, while a random set of sequence gives around $3 \cdot l/4 = 57$. Surely this value can never be reached in a flow reactor with a replication accuracy near 1, because every new sequence is inherited from an item of the actual population and is therefore identical or minimal different from its ancestor in most of the cases. But what about the maximal Hamming distance of any two sequences in a population dump? If independent clusters exist which spread, one would expect that relatively distant regions of sequence space are explored. The figure 3.21 shows that the average diameter of a population dump is only Hamming distance 34.2. This indicates that clusters have a limited lifetime and have to split off from other clusters frequently.

Is the expected survival time of a cluster series dependent on its population size? To explore this question we took all cluster series that disappeared through die out, calculated its maximal population size and ploted it against the number of replications during their existance (figure 3.22). We excluded cluster series ending in unions and exchanges, because they are arbitrary cluster series termination reasons. As expected only for short maximal population sizes conclusions on the lifetime can be drawn. For larger population sizes the difference between late and early dieout grows exponentially.

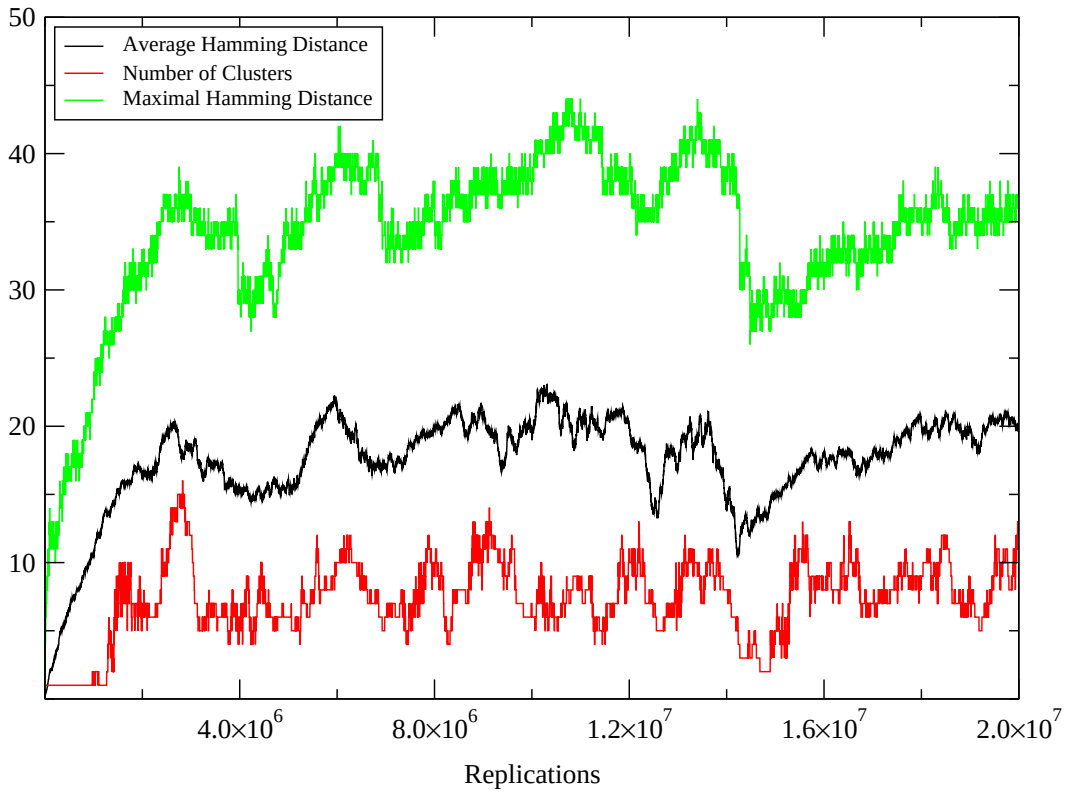



Figure 3.21: Maximal and average Hamming distance and number of clusters in a flow reactor run without fitness increases. After the population spreads in sequence space, these values saturate. The maximal and average diameter of the mutant cloud is limited by the population size of the reactor. Due to their limited lifetime, indepenent moving clusters doesn't exist. New clusters split off from existing clusters and therefore they are located close togehter in sequence space.

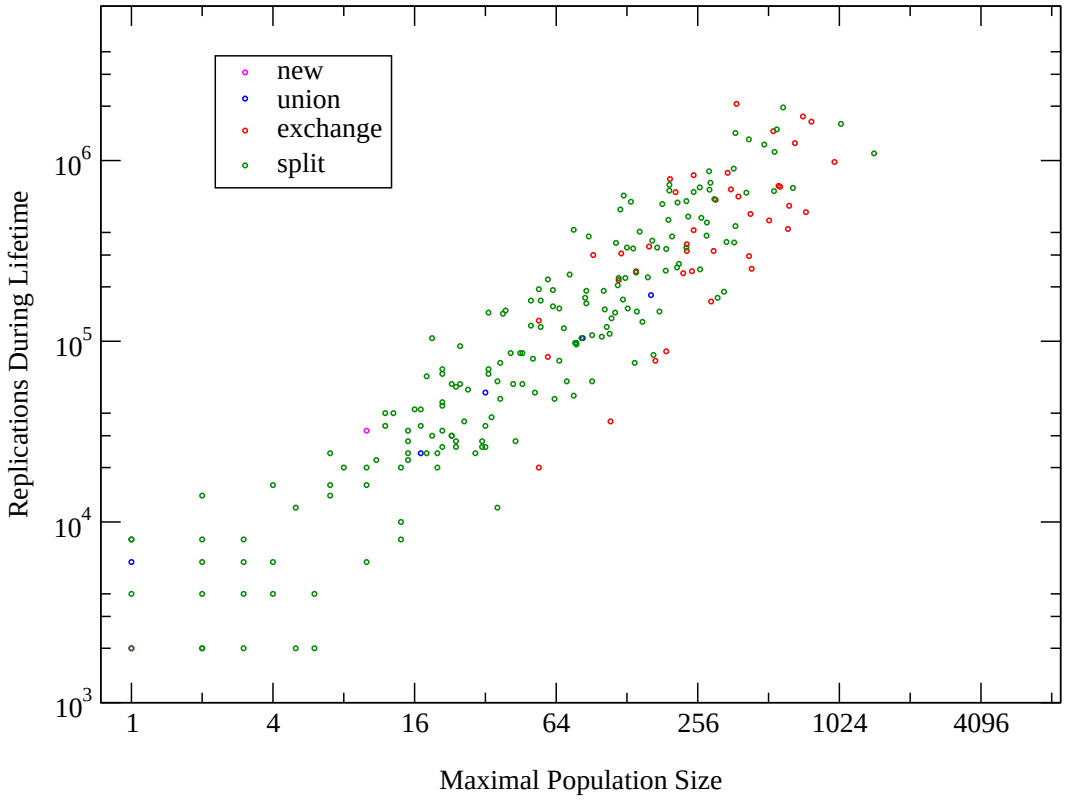



Figure 3.22: A log/log plot of the maximal population size of a cluster series vs. the replications during its lifetime. All cluster series in this plot become extinct trought die out (type (e)). The different colours show the origin of the cluster series. The larger the maximal population size, the less one can foretell about its lifetime.

# 4 A Comprehensive Model of Evolution

In this chapter the creation of a new sequence in a flow reactor which is not present yet is described. Dependent on the share $x_j$ and the fitness $f_j$ of any present sequence $I_j$ in the reactor, the new sequence $I_\ell$ is created with a frequency $Q_{\ell j}$. In principle the sequence $I_\ell$ can be created out of any RNA sequence present at the current moment. Since the mutation rate is usually low (e.g. $p = 0.001$) it is rather unlikely (but possible) that the Hamming distance between $I_j$ and $I_\ell$ is greater than than 1 or 2.

$$\frac{dx_\ell}{dt} = \sum_{j=1,j\neq\ell}^{n} Q_{\ell j} f_j x_j$$

The sequence $I_\ell$ is then folded into its structure $S_\ell$. In an *in silico* experiment $S_\ell$ can be the secondary structure folded with an appropriate folding algorithm like the one implemented in the Vienna RNA Package [25, 52]. *In vitro* $S_\ell$ represents the 3-dimensional structure of the biomolecule.

$$S_\ell = \Psi(I_\ell)$$

The new sequence is evaluated and a fitness value based on the secondary structure is assigned. In this work the following fitness function for sequences with length $l$ is used:

$$f_\ell = \frac{1}{0.01 + d_{\ell t}^h / l} \tag{4.1}$$

Any distance measure $d_{\ell t}$ between the shape $S_\ell$ and the target shape $S_t$ can be applied, like the Hamming distance between the parenthesis notations of these two structures.

*In vitro* the fitness can for example be based on the binding to another molecule, the speed of replication, or an enzymatic function.

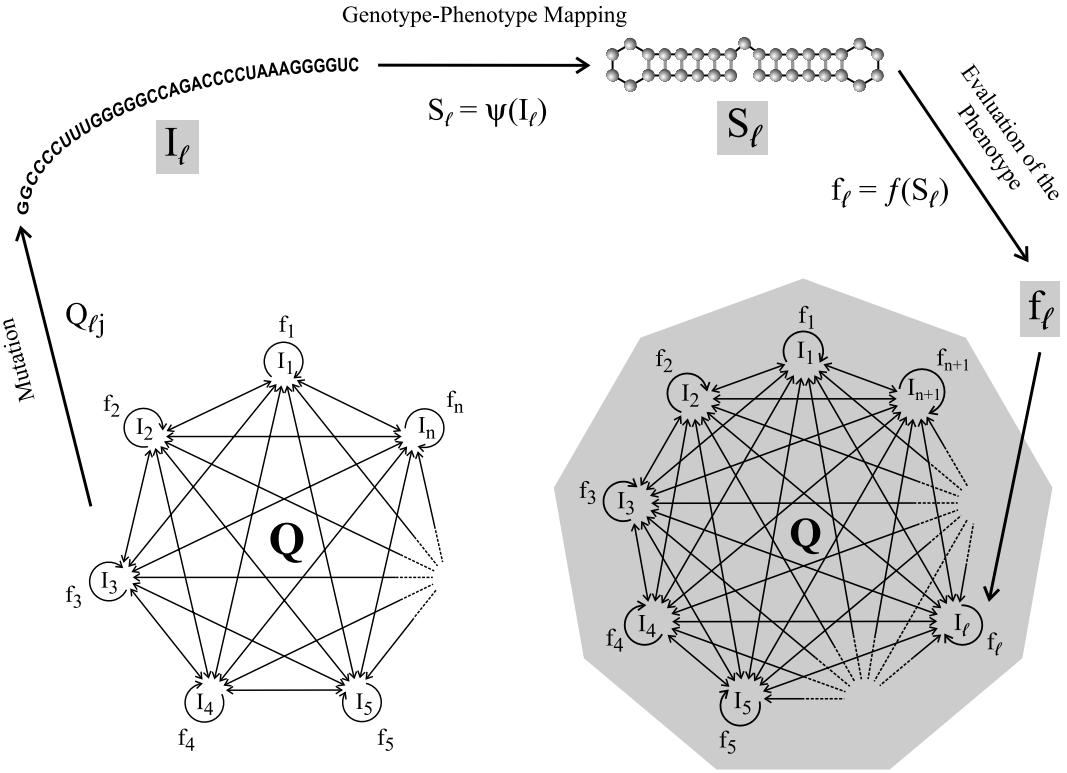



Figure 4.1: A comprehensive model of evolution: After the creation of sequence $I_\ell$ with a frequency $Q_{\ell j}$ through mutation of any sequence $I_j$, the folding to its minimal free energy (mfe) structure $S_\ell$ and its fitness rating $f_\ell$, the sequence becomes present in the flow reactor with a single copy. Its further destiny is dependent on the flow rate, its fitness, the probability to be reinvented by any other (neighbouring) sequences and on chance of cause.

Within the flow reactor population the new sequence $I_\ell$ has to compete with the rest of the population to increase its population share $x_\ell$. This increase can be an exact replication ($dx_\ell = (Q_{\ell\ell}\, f_\ell\, x_\ell)\, dt$) or the mutation to $I_\ell$ of any other sequence ($dx_\ell = \left(\sum_{j=1,j\neq\ell}^{n} Q_{\ell j} f_j\, x_j\right) dt$). On the other hand sequences are eliminated regardless of their fitness with the current flow rate ($dx_\ell = [x_\ell \Phi(t)] dt$). The change of the share $x_\ell$ sums up to:

$$\frac{dx_\ell}{dt} \;=\; Q_{\ell\ell}\; f_\ell\; x_\ell \;-\; x_\ell\; \Phi(t) + \sum_{j=1,j\neq\ell}^{n} Q_{\ell j} f_j\, x_j$$

According to the quasispecies theory [6, 35] a cloud of sequences is formed in the near neighbourhood of the dominating sequences, which are mainly products of erroneous replications of the dominating sequences. Mutational backflow from this sequence cloud increases the population size of the dominating sequences, which is important for their survival. Due to mutations the outflow of a sequence in a (nearly) fitness homogeneous population is always bigger than its error-free replication rate. Nevertheless such a cloud around the dominating sequence can give them the crucial advantage and can increase the probability for survival for some time [1, 54]. But according to Motoo Kimura's *neutral mutation-random drift hypothesis* [29] the movement in sequence space continues, which generates new dominating sequences after some time.

# 5 Conclusions and Outlook

Evolutionary optimization of RNA structures in simulated flow reactors is a method to explore the mechanisms of evolution in general. Some thousand (simulated) RNA molecules are optimized by means of an algorithm to compute individual trajectories for chemical reaction mechanisms conceived by Daniel Gillespie [18, 19]. The RNA sequences are folded into their minimum free energy (mfe) secondary structures and their fitness is rated using the Hamming distance to a target shape [14]. The flow reactor run is terminated as soon as the target structure is found and settled. Then the *relay series* is calculated which is a kind of backtracking from the target structure to a shape of the start population on the level of structures. Beside some generic properties of folding ((i) more sequences than structures, (ii), some structures are more frequent than others, (iii) neutral networks, (iv) shape space covering) [15, 16] other features of the optimization process are known: (i) Fast periods containing cascades of adaptive changes are interrupted by long quasi-stationary epochs of neutral evolution, during which populations drift randomly on neutral networks. (ii) Continuous transitions are caused by single point mutations leading to a different but frequent shape in the neighbourhood of the template's structure. If a new structure is created through a mutation which causes major structural changes and is rarely part of the neighbourhood of the template's structure it is called a discontinuous transition. Usually a discontinuous transition is attended by a concurrent fitness increase (major transition).

Statistics of the trajectories shows that every flow reactor optimization with identical start population leads to a different *relay series* track to the target shape. In other words, no trajectory has been reproduced in detail (By reproduction we mean that the same trajectory is obtained for identical inital conditions except different start conditions of the pseudo random number generator). Thus neither the succession of structures nor the length of the

quasi-stationary plateaus can be predicted. This leads to a vast scatter in the number of replications of flow reactor runs.

Reactors with a larger population size have two important differences: (i) The number of continuous transitions is significantly smaller in contrast to the number of discontinuous transitions, which stays astonishing constant with varying population size. (ii) It takes more replications to find the target structure.

In the flow reactor simulation discontinuous transitions are usually rare and unique events. If the structures of the involved sequences are part of the *relay series* they form a bottleneck to the future population of the reactor. Any fit sequence created after the discontinuous transition is inherited from them. If we are also concerned with a major transition the population center shifts from the center of the former population to the sequence created through the discontinuous transition.

In general, serial transfer experiments perform worse than flow reactors. The larger the reduction step and maximal population size the more replications are needed to find the target shape.

*Lineages* are an inheritance series on the level of sequence copies. Folded into their mfe structures the comparison to *relay series* shows a conformity during discontinuous transitions but different paths during continuous transitions on the fitness plateaus.

It could be shown that during a phase of constant fitness the population are lumped together in distinct clusters. Graphically this was done by a principal component analysis (PCA). Numerically the subgraph-clustering revealed that nearly no sequence exchange between clusters can be observed. Similar to in vitro experiments on fitness plateaus large genotypic but little phenotypic changes are found. During the fast adaptive phase at the beginning of the run the situation is inverted: Little changes on the primary sequence are accompanied by large structural changes.

With the *Flow Reactor Class Library* programs for the simulation and analysis of *in silico* flow reactor runs are provided. Written in C++ the classes are programmed modular and are easily extensible.

Discontinuous transitions play a crucial role in evolutionary optimization. Although the sequences which can perform a discontinuous transition are not distinguishable in terms of fitness they allow the exploration of new and often fitter neutral networks, whose neighbourhood can contain structures much closer to the target shape. The number of discontinuous transitions stay constant with varying population size and seems to be dependent on the size and structure of the target as well as the fitness function and the complexity of the folding algorithm.

The fitness function used in this work is based on the Hamming distance to the target shape. In future experiments the folding with h-type pseudo knots [23, 24], the suboptimal structures [55], the results of kinetic folding, or three-dimensinal interactions could be included into the fitness function to obtain results, which are more comparable to natural processes. The *Flow Reactor Class Library* should be extended with a graphical user interface (GUI), a communication interface, and a database for the administration and evaluation of the results.

# Bibliography


[1] C. Burch and L. Chao. Evolvability of an RNA virus is determined by its mutatioal neigbourhood. *Nature*, 406:625–628, 2000.

[2] J. Cupal, S. Kopp, and P. F. Stadler. RNA shape space topology. *Artificial Life*, 6:3–23, 2000.

[3] J. Cupal, P. Schuster, and P. F. Stadler. Topology in phenotype space. In *Computer Science in Biology*, GCB'99 Proceedings, pages 9–15. Univ. Bielefeld, Hannover, DE, 1999.

[4] B. Derrida and L. Peliti. Evolution in a flat fitness landscape. *Bull. Math. Biol.*, 53:355–382, 1991.

[5] M. Eigen. Selforganization of matter and the evolution of biological macromolecules. *Die Naturwissenschaften*, 10:465–523, 1971.

[6] M. Eigen, J. McCaskill, and P. Schuster. The molecular quasispecies. *Adv. Chem. Phys.*, 75:149 – 263, 1989.

[7] M. Eigen and P. Schuster. The hypercycle. A principle of natural self-organization. Part A: Emergence of the hypercycle. *Naturwissenschaften*, 64:541–565, 1977.

[8] M. Eigen and P. Schuster. The hypercycle. A principle of natural self-organization. part B: The abstract hypercycle. *Naturwissenschaften*, 65:7–41, 1978.

[9] M. Eigen and P. Schuster. The hypercycle. A principle of natural self-organization. part C: The realistic hypercycle. *Naturwissenschaften*, 65:341–369, 1978.

[10] M. Eigen and P. Schuster. *The Hypercycle: A principle of natural self-organization.* Springer, Berlin, 1979.

[11] S. F. Elena, V. S. Cooper, and R. E. Lenski. Punctuated evolution caused by selection of rare beneficial mutants. *Science*, 272:1802–1804, 1996.

[12] C. Flamm, W. Fontana, I. L. Hofacker, and P. Schuster. RNA folding at elementary step resolution. *RNA*, 6:325–338, 2000.

[13] W. Fontana, D. A. M. Konings, P. F. Stadler, and P. Schuster. Statistics of RNA secondary structures. *Biopolymers*, 33:1389–1404, 1993.

[14] W. Fontana and P. Schuster. A computer model of evolutionary optimization. *Biophys. Chem.*, 26:123–147, 1987.

[15] W. Fontana and P. Schuster. Continuity in evolution: On the nature of transitions. *Science*, 280:1451–1455, 1998.

[16] W. Fontana and P. Schuster. Shaping space: The possible and the attainable in RNA genotype-phenotype mapping. *J. Theor. Biol.*, 194:491–515, 1998.

[17] J. H. Gaddum. Lognormal distributions. *Nature*, 156:463–466, 1945.

[18] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.

[19] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81:2340–2361, 1977.

[20] N. S. Goel and N. Richter-Dyn. *Stochastic Models in Biology.* Academic Press, New York, 1974.

[21] W. Grüner, R. Giegerich, D. Strothmann, C. Reidys, J. Weber, I. L. Hofacker, P. F. Stadler, and P. Schuster. Analysis of RNA sequence structure maps by exhaustive enumeration. I. Neutral networks. *Mh.Chem.*, 127:355–374, 1996.

[22] R. W. Hamming. Error detecting and error correcting codes. *Bell Syst. Tech. J.*, 29:147–160, 1950.

[23] C. Haslinger. *Prediction Algorithms for Restricted RNA Pseudoknots.* PhD thesis, Univ. for Vienna, 2001.

[24] C. Haslinger and P. F. Stadler. RNA structures with pseudo-knots: Graph-theoretical, combinatorical, and statistical properties. *Bul. Math. Biol.*, 1:1–33, 1998.

[25] I. L. Hofacker, W. Fontana, P. F. Stadler, S. Bonhoeffer, M. Tacker, and P. Schuster. Fast folding and comparison of RNA secondary structures. *Mh. Chem.*, 125:167–188, 1994.

[26] J. H. Holland. *Adaption in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology.* University of Michigan Press, 1975.

[27] M. A. Huynen, P. F. Stadler, and W. Fontana. Smoothness within ruggedness: The role of neutrality in adaptation. *Proc. Natl. Acad. Sci. USA*, 93:397–401, 1996.

[28] J. A. Jaeger, D. H. Turner, and M. Zuker. Improved predictions of secondary structures for RNA. *Proc. Natl. Acad. Sci. USA*, 86:7706–7710, 1989.

[29] M. Kimura. *The Neutral Theory of Molecular Evolution.* Cambridge University Press, Cambridge, UK, 1983.

[30] J. R. Koza, F. H. Bennett, M. A. Keane, and D. Andre. *Genetic Programming III: Darwinian Invention and Problem Solving.* Morgan Kaufmann, 1998.

[31] R. E. Lenski and M. Travisano. Dynamics of adaptation and diversification: A 10 000-generation experiment with bacterial populations. *Proc. Natl. Acad. Sci. USA*, 91:6808–6814, 1994.

[32] M. Matsumoto and T. Nishimura. Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. on Modeling and Computer Simulation*, 8:3–30, 1998.

[33] D. R. Mills, R. L. Peterson, and S. Spiegelman. An extracellular Darwinian experiment with a self-duplicating nucleic acid molecule. *Proc. Natl. Acad. Sci. USA*, 58:217–224, 1967.

[34] E. V. Nimwegen, J. P. Crutchfield, and M. Huynen. Neutral evolution of mutational robustness. *Proc. Natl. Acad. Sci. USA*, 96:9716–9720, 1999.

[35] M. A. Nowak. What is a quasispecies. *Trends Ecol. Evol.*, 7:118–121, 1992.

[36] D. Papadopoulos, D. Schneider, J. Meier-Eiss, W. Arber, R. E. Lenski, and M. Blot. Genomic evolution during a 10 000-generation experiment with bacteria. *Proc. Natl. Acad. Sci. USA*, 96:3807–3812, 1999.

[37] S. K. Park and K. W. Miller. Random number generators: Good ones are hard to find. *Communications of the ACM*, 31(10):1192–1201, 1988.

[38] P. J. Plauger, A. A. Stepanov, M. Lee, and D. R. Musser. *The C++ Standard Template Library.* Prentice Hall, 2000.

[39] I. Rechenberg. *Evolutionsstrategie: Optimierung technisher Systeme nach Prinzipien der biologischen Evolution.* Friedrich Frommann Verlag, 1973.

[40] C. Reidys, C. Forst, and P. Schuster. Replication and mutation on neutral networks. *Bull. Math. Biol.*, 63:57–94, 2001.

[41] R. Robson. *Using the STL: The C++ Standard Template Library.* Springer, 1997.

[42] P. Schuster. How to search for RNA structures. theoretical concepts in evolutionary biotechnology. *J. Biotechnology*, 41:239–257, 1995.

[43] P. Schuster. Genotypes with phenotypes: Adventures in an RNA toy world. *Biophys. Chem.*, 66:75–110, 1997.

[44] P. Schuster. Landscapes and molecular evolution. *Physica D*, 107:351–365, 1997.

[45] P. Schuster and W. Fontana. Chance and necessity in evolution: Lessons from RNA. *Physica D*, 133:427–452, 1999.

[46] P. Schuster, W. Fontana, P. F. Stadler, and I. L. Hofacker. From sequences to shapes and back: A case study in RNA secondary structures. *Proc. Roy. Soc. (London) B*, 255:279–284, 1994.

[47] P. Schuster and K. Sigmund. Dynamics of evolutionary optimization. *Ber. Bunsenges. Phys. Chem.*, 89:668–682, 1985.

[48] P. Schuster and A. Wernitznig. Stochastic dynamics of neutral evolution RNA in the flow reactor. *Proc. Natl. Acad. Sci.*, submitted, 2001.

[49] S. Spiegelman. An approach to the experimental analysis of precellular evolution. *Quart. Rev. Biophys.*, 4:213–253, 1971.

[50] B. M. Stadler, P. F. Stadler, G. P. Wagner, and W. Fontana. The topology of the possible: Formal spaces underlying patterns of evolutionary change. *J. Theor. Biol.*, 2000. Submitted.

[51] L. Wall, T. Christiansen, and J. Orwant. *Programming Perl*. O'Reilly, 2000.

[52] A. E. Walter, D. H. Turner, J. Kim, M. H. Lyttle, P. Müller, D. H. Mathews, and M. Zuker. Co-axial stacking of helixes enhances binding of oligoribonucleotides and improves predictions of RNA folding. *Proc. Natl. Acad. Sci. USA*, 91:9218–9222, 1994.

[53] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, pages 236–244, 1963.

[54] C. O. Wilke, J. L. Wang, C. Ofria, R. E. Lenski, and C. Adami. Evolution of digital organisms at high mutation rates leads to survival of the flattest. *Nature*, pages 331–333, 2001.

[55] S. Wuchty, W. Fontana, I. L. Hofacker, and P. Schuster. Complete Suboptimal Folding of RNA and the Stability of Secondary Structures. *Biopolymers*, pages 145–165, 1998.

[56] G. Zipf. *Human Behaviour and the Principle of Least Effort.* Addison-Wesley, Reading, MA, 1949.

[57] M. Zuker and P. Stiegler. Optimal computer folding of larger RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9:133–148, 1981.

# Curriculum vitae

## Andreas Wernitznig

* 8. Mai 1969 in Klagenfurt

verheiratet, eine Tochter

| | | |
|---|---|---|
| 1975 - 1979 | Volksschule der Ursulinen | Klagenfurt |
| 1979 - 1983 | Bundesgymnasium Völkermarkterring | Klagenfurt |
| 1983 - 1987 | Bundesoberstufenrealgymnasium | Klagenfurt |
| 1987 | Reifeprüfung mit gutem Erfolg | Klagenfurt |
| 1988 | Zivildienst | Klagenfurt |
| 1988 - 1997 | Studium der Lebensmittel und Biotechnologie an der Universität für Bodenkultur | Wien |
| 1995 - 1996 | Diplomarbeit am Institut für Tumorbiologie und Krebsforschung der Universität Wien in der Arbeitsgruppe Molekulare Genetik „Charakterisierung des Proteins Scp160p aus Saccharomyces cerevisiae“ | Wien |
| 11/1997 | Sponsion zum Dipl. Ing. | Wien |
| seit 2/2001 | Geschäftsführer der Insilico Software GmbH | Wien |
| 1997 - 2001 | Dissertation am Institut für theoretische Chemie | Wien |