bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

SilentMutations (SIM): a tool for analyzing long-range RNA-RNA interactions in viral genomes and structured RNAs

Daniel Desirò[1,2], Martin Hölzer[1,2], Bashar Ibrahim[1,2], Manja Marz[1,2,3]

[1]European Virus Bioinformatics Center, Jena, Germany
[2]RNA Bioinformatics and High-Throughput Analysis Jena, Friedrich Schiller University Jena, Jena, Germany
[3]Leibniz Institute for Age Research-Fritz Lipmann Institute, Jena, Germany

ABSTRACT

Background: A single nucleotide change in the coding region can alter the amino acid sequence of a protein. In consequence, natural or artificial sequence changes in viral RNAs may have various effects not only on protein stability, function and structure but also on viral replication.
In recent decades, several tools have been developed to predict the effect of mutations in structured RNAs such as viral genomes or non-coding RNAs. Some tools use multiple point mutations and also take coding regions into account. However, none of these tools was designed to specifically simulate the effect of mutations on viral long-range interactions.
Results: Here, we developed `SilentMutations` (`SIM`), an easy-to-use tool to analyze the effect of multiple point mutations on the secondary structures of two interacting viral RNAs. The tool can simulate disruptive and compensatory mutants of two interacting single-stranded RNAs. This allows a fast and accurate assessment of key regions potentially involved in functional long-range RNA-RNA interactions and will eventually help virologists and RNA-experts to design appropriate experiments.
`SIM` only requires two interacting single-stranded RNA regions as input. The output is a plain text file containing the most promising mutants and a graphical representation of all interactions.
Conclusion: We applied our tool on two experimentally validated influenza A virus and hepatitis C virus interactions and we were able to predict potential double mutants for *in vitro* validation experiments.
Availability: The source code and documentation of `SIM` are freely available at `github.com/desiro/silentMutations`.

Key words: Codon Mutation, Virology, RNA virus, Virus Bioinformatics, RNA secondary structure, double-mutant, silent mutation

Introduction

In the last decades, several computational tools for the analysis of RNA secondary structures have been developed. However, tools specifically targeting virus needs are still rare and under-researched[1–3]. For example, long-range RNA-RNA interactions (LRIs) play an important role in the life cycle of RNA viruses.
Many LRIs are already known to act directly as activators or inhibitors of viral replication and translation[4]. For example, several interactions have been recently identified as potential new LRIs in the genome of the hepatitis C virus (HCV) using `LRIscan`[5]. Some of these computationally identified interactions have already been verified experimentally[6–13].

Apart from long-range interactions forming in a single sequence (intra-molecular structures), LRI-like structures can also occur between two separate sequences, for example between two viral segments. Whereas the general function of such interactions is still under investigation, they can be seen as viral LRIs and are presumably responsible for the correct packaging of all segments into the viral capsid of segmented viruses such as the influenza A virus (IAV)[14–17].
Since the number of known and possibly functional LRIs is growing rapidly, it is essential to have an effective verification method. Technically, such interactions can be destroyed by removal or mutation of the interacting segments using *in vitro* experiments[8,18]. Such sequence changes can result in secondary structure changes and finally

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

manifest different viral titers. However, with the alteration of the sequence, unwanted effects can arise that not only effect the LRI.
To cope with this issue, we can modify both interacting RNA segments simultaneously. Additionally, we aim for the combination of both mutated segments to result in a similar interaction strength in comparison to the interaction between the wild type (WT) sequences. A single mutated sequence segment in combination with the respective WT sequence should destroy the interaction. Such a technique was recently used to verify a possible LRI between two IAV segments[15].
Several computational tools are available that can alter a sequence and report alternative secondary structures with one or even multiple mutations[19–26]. However, these tools are designed for specific applications and do not meet the requirements of the experimental technique described above: the combinatorial *in vitro* verification of RNA-RNA interactions.
In this study, we present a tool called `SilentMutations` (`SIM`) that effectively simulates synonymous (silent) compensatory mutations in two single-stranded viral RNAs and is therefore appropriate for the *in vitro* assessment of predicted LRIs.

Materials and Methods

Here, we present a command-line tool, called `SilentMutations` (`SIM`), that can simulate synonymous structure-disrupting and structure-preserving mutation pairs within coding regions for long-range RNA-RNA interaction experiments. The tool has been written in `Python` (v3.6.5) and heavily relies on the `RNAcofold` python site-package of the `ViennaRNA Package` (v2.4)[27]. As the `ViennaRNA Package` is available for Linux, Windows, and MacOS, all three different platforms are supported by `SIM`. The various parameters of `SIM` are fully adjustable and several filters (**Fig. 1**) allow a fast and accurate prediction, therefore the tool can be run on a standard notebook.
A simple run takes about 10 seconds when using the standard parameters, two sequences of length 21 as input, and four cores. However, increasing the length of the two sequences to 30 nt already results in a runtime of 1 minute and 10 seconds. Generally, the runtime highly depends on the parameter setup and the length of the input sequences.
For two nucleotide sequences of length k and l and the number of possible synonymous codons c, the worst case runtime can be described by $\mathcal{O}(c^{\frac{k}{3}+\frac{l}{3}})$, with c between 1 and 6 (average: $\frac{64}{21} \approx 3.048$). This large search space can be drastically reduced by various filter steps implemented in `SIM` (see below).
Because the runtime depends exponentially on the length of the input sequences, we recommend to limit the interaction length to maximal 30 nt. Additionally, longer sequences are more likely to form intra-molecular secondary structures instead of forming stable structures with each other. Whereas `SIM` can be used to evaluate longer sequences, the tool is originally intended for two smaller sequence segments forming a stable secondary structure. For the prediction of LRIs in viral RNA genomes and longer sequences, we recommend `LRIscan`[5]. The main computational steps of `SIM` are (1) preprocessing, (2) permutation, (3) attenuation, (4) recovery, and (5) sampling.
A particular challenge is posed by negative single-stranded RNA viruses such as Ebola, Marburg, and Coronaviruses, as most databases only contain the positive strand of the virus genome. While the protein-coding sequence is encoded on the positive strand, the folding of these viral RNAs happens on the negative strand. Depending on the user input, `SIM` can automatically create the reverse-complement strand of negative single-stranded RNA viruses for folding, while maintaining codon integrity on the positive strand. This can be easily achieved with the `-virusclass=ssRNA-`, `-reverse`, and `-complement` options. The user can directly provide the positive strand sequence from the database in a fasta file, with the reading frame and regions on the positive strand.
The default output of `SIM` prints all relevant information of the folding in a plain text file and additionally creates for each folding a `VARNA`[28] command for visualization. Additionally, if the tool can find installations of `VARNA` and `inkscape`[29], it will directly generate high-quality vector graphics of the various foldings. A default binary of `VARNA` (v3.93) is provided with the tools source code.
`SIM` is primarily designed for two sequence snippets (hereinafter referred to as snips) in coding regions, but can also be used to simulate interactions between coding and non-coding regions and for interactions between two non-coding regions. This can be done with the `-noncoding1` and `-noncoding2` parameters, to define the first or second sequence as non-coding, respectively. This makes the prediction of interactions between RNA molecules with selection pressure on the sequence for translation and RNA molecules with selection pressure on secondary structures possible.

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

We use the following notation for the minimum free energy (*mfe*) obtained by folding sequence *x* with sequence *y* via `RNAcofold`:

$$mfe(x, y)$$

The *mfe* is represented by a negative value in $\frac{kcal}{mol}$ and therefore having a lower *mfe* results in a more stable structure.

Preprocessing

In the first step of `SIM`, sequence snips are extracted from longer sequence templates while keeping codons intact (**Fig. 1**). Especially, the difficult extraction of snips in the exact reading frame from negative sense single-stranded RNAs can be easily handled by the build-in functionalities of the tool. Thus, `SIM` directly accepts the whole sequence as input, together with a start and end position for the snips and the reading frame. The tool first extracts the requested interacting segments from each sequence. Each of them will be automatically increased in both directions if the provided range would otherwise split a codon, based on the predefined reading frame (**Fig. 1**). This ensures a preservation of all codons. Both sequences, *pre-snip1* (ps_1) and *pre-snip2* (ps_2), will then be fold together using `RNAcofold` to acquire only codons involved in the folding. Unpaired codons at the termini will be removed to get the final *snip1* (s_1) and *snip2* (s_2) from which `SIM` calculates the two mutant sequences *mut1* (m_1) and *mut2* (m_2).

Permutation

In this step, all possible permutations *pre-perms1* (*P**) and *pre-perms2* (*Q**) of synonymous codons are created from each sequence in order to find the most suitable mutations that maximize the distance between $mfe(s_1,s_2)$ and $mfe(s_1,m_2)$ and the distance between $mfe(s_1,s_2)$ and $mfe(m_1,s_2)$, while keeping the wild-type $mfe(s_1,s_2)$ and the double-mutant $mfe(m_1,m_2)$ similar. The number of these permutations can be vast, depending on the number of synonymous codons and the length of the sequence. To reduce the computational complexity, this step also removes every permutation sequence with more mutations than predefined by the `-mutations` parameter resulting in the sets *perms1* (*P*) and *perms2* (*Q*), see **Fig. 1**. With our implementation of this parameter, we aim to keep the number of introduced mutations small, while simultaneously maximizing the *mfe* difference between wild-type and single mutants.

Attenuation

In this step, the computational complexity is further reduced and valid sequences with reduced *mfe* folding scores are created. Importantly, having a small *mfe* difference between the wild-type ($mfe(s_1,s_2)$) and the single mutants ($mfe(s_1,m_2)$ or $mfe(m_1,s_2)$) would diminish the mutational effect on the interaction. It is therefore vital to increase the distance between these *mfe* scores. Therefore, the user can directly specify the required distance with the filter percentage (`-filterperc`) parameter. The defined value is then multiplied with the $mfe(s_1,s_2)$ value to obtain an upper limit (denoted *l*) for $mfe(s_1,m_2)$ and $mfe(m_1,s_2)$. Then, the tool creates all foldings between s_1 and *Q* as well as *P* and s_2. Each folding $mfe(s_1, q_j \in Q)$ and $mfe(p_i \in P, s_2)$ (with *i* and *j* defining the i-th and j-th element of set *P* and *Q*, respectively) is only considered for the next recovery step if the *mfe* is higher than *l*. The new sets *mut1* (*U*) and *mut2* (*V*) have a drastically reduced size compared to *P* and *Q* which greatly reduces the computational complexity for finding the two mutant sequences m_1 and m_2 where $mfe(m_1,m_2)$ is similar to the wild-type $mfe(s_1,s_2)$. While it seems to be convenient to set a very low `-filterperc` parameter, this could also result in an empty permutation set. On the other hand, setting a high `-filterperc` could result in longer running times. It is therefore recommended to start with the default *0.5* and later adjust this parameter depending on the performance.

Recovery

To find double-mutants (mut) with a similar folding *mfe* such as the wild-type (WT) folding *mfe*, each sequence $u_i \in U$ and $v_i \in V$ has to be folded against each other with `RNAcofold`. The running time of this folding process strongly depends on the previous attenuation step. When using the unprocessed *P** and *Q** sets from the example in **Fig. 1**, the number of required folding operations would be around 24 million and reduced to only 37,000 for the filtered sets *U* and *V*. Some LRIs function as inhibitors for viral replication[4] and a stronger interaction would decrease functional viral reproduction. For the simulation of the double-mutant interaction it is therefore necessary to set a lower but also an upper similarity threshold. Each of the calculated $mfe(u_i \in U, v_j \in V)$ values are therefore compared with the wild-type

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

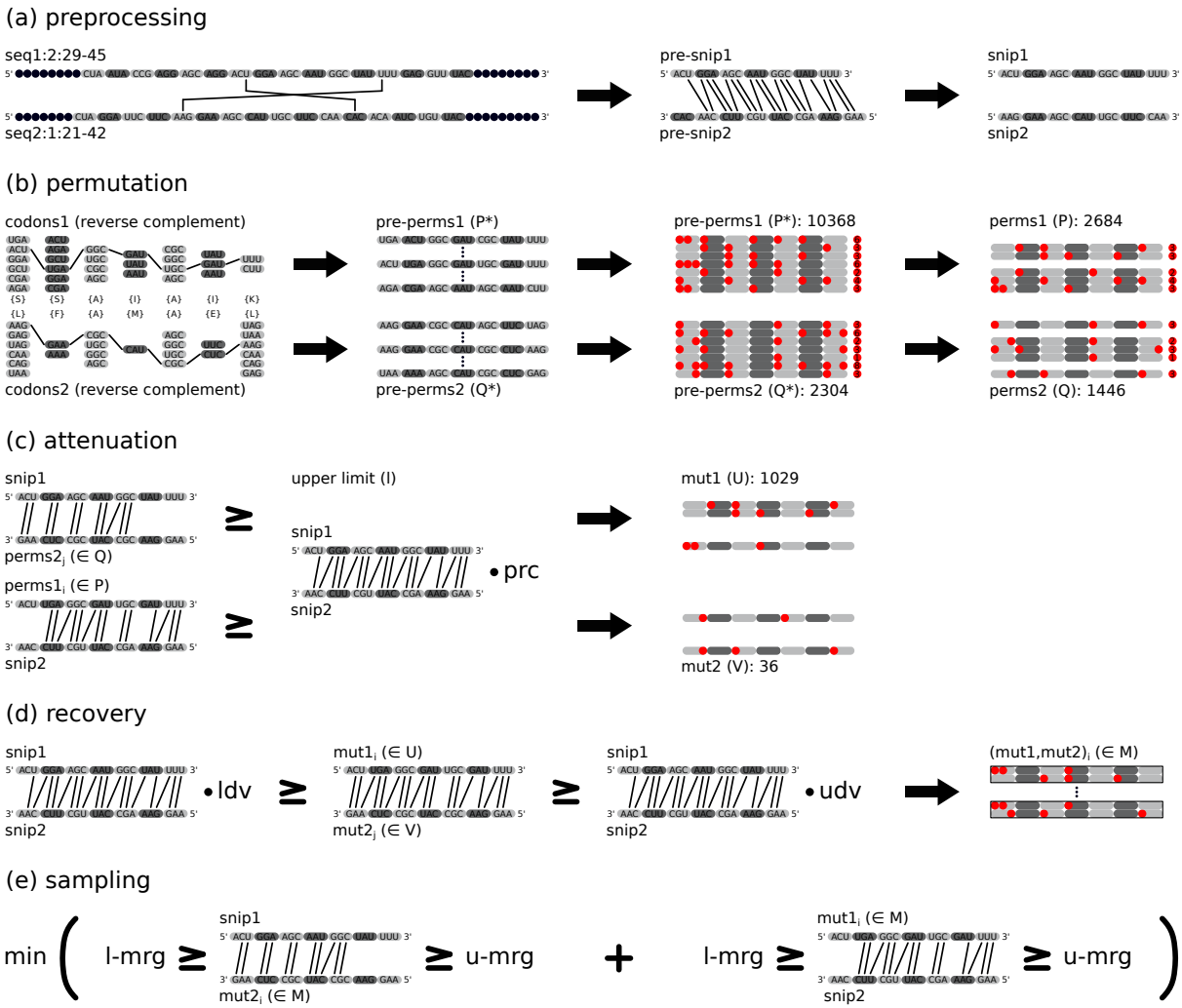

Figure 1: Overall workflow of `SIM`, exemplarily shown for two sequences from a negative single-stranded RNA virus genome (ssRNA-) (a) The first step extracts the defined range in each sequence and possibly increases the range to preserve codons based on the given reading frame. Both so-called *pre-snips* are folded with `RNAcofold` to remove unpaired codons at the endings. We refer to an extracted single-stranded RNA snippet as *snip*. **(b)** All possible codon permutations (here, reverse complements due to ssRNA-) are generated for both *snips* (called *perms*) to conserve the amino acid sequence. Permutations with too many mutations defined by the mutations (`-mut`) parameter are discarded. **(c)** To further decrease the total number of combinations, `SIM` will fold each *snip* with all permutations of the other *snip* and keep permutations with a *mfe* higher or equal to *mfe*(*snip1,snip2*) times the filter percentage (`-prc`) parameter, denoted as upper limit *l*. **(d)** All remaining *snip1* permutations will then be folded against all *snip2* permutations to find *snip1* and *snip2* mutations with a similar fold *mfe*. A double-mutant *mfe*(*mut1,mut2*) fold is considered to be similar, if its *mfe* is lower than the wild-type *mfe*(*snip1,snip2*) times the lower deviation (`-ldv`) parameter and higher than *mfe*(*snip1,snip2*) times the upper deviation (`-udv`) parameter. **(e)** The last step will minimize a combined single-mutant *mfe*(*snip1,mut2*) and *mfe*(*mut1,snip2*) by keeping both mutations in a similar range depending on the mutation range (`-mrg`) parameter. The range for each mutation combination is defined by $(\mathit{mfe}(\mathit{snip1}, \mathit{mut2}) + \mathit{mfe}(\mathit{mut1}, \mathit{snip2})) \cdot 0.5 \cdot (1 + \texttt{-mrg})$ for the lower threshold (l-mrg) and $(\mathit{mfe}(\mathit{snip1}, \mathit{mut2}) + \mathit{mfe}(\mathit{mut1}, \mathit{snip2})) \cdot 0.5 \cdot (1 - \texttt{-mrg})$ for the upper threshold (u-mrg). Different sets and their abbreviations are given in brackets. Details can be found in the Methods.

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

mfe(s_1,s_2). A combination of two sequences *u* and *v* is only valid if

$$\textit{mfe}(s_1, s_2) \cdot \texttt{ldv} \geq \textit{mfe}(u, v) \geq \textit{mfe}(s_1, s_2) \cdot \texttt{udv}$$

with `ldv` as the lower deviation (`-lowerdev`) and `udv` as the upper deviation (`-upperdev`) parameter. Valid (u, v) double-mutant pairs are stored as tuples in set *M*.

Sampling

Now, all pairs in the final set *M* contain only mutated sequence pairs with the desired properties. However, greedily selecting the i-th tuple $(u, v)_i \in M$ that minimizes

$$\textit{mfe}(s_1, v_i \in M) + \textit{mfe}(u_i \in M, s_2)$$

could result in unfavorable interaction strengths between WTs and mutants. An unfavorable *v* and *u* pair would result in a significantly higher single-mutant *mfe* of the mutant folded with the WT sequence in comparison to the *mfe* of the other mutant and WT combination. Therefore, it is recommended to keep both, *mfe*(s_1,$v_i \in M$) and *mfe*($u_i \in M, s_2$), in a similar range while minimizing the combined *mfe*. Accordingly, the sampling step takes the mutation range (`-mrg`) parameter into account and calculates the lower threshold `l-mrg` as

$$(\textit{mfe}(s_1, v_i \in M) + \textit{mfe}(u_i \in M, s_2)) \cdot 0.5 \cdot (1 + \texttt{mrg})$$

and the upper threshold `u-mrg` as

$$(\textit{mfe}(s_1, v_i \in M) + \textit{mfe}(u_i \in M, s_2)) \cdot 0.5 \cdot (1 - \texttt{mrg})$$

for each mutant tuple. The most suitable mutant pair, holding all predefined requirements, is finally found by calculating:

$$\textit{min}(\textit{mfe}(s_1, v_i \in M) + \textit{mfe}(u_i \in M, s_2))$$

with

$$\texttt{l-mrg} \geq \textit{mfe}(s_1, v_i \in M) \geq \texttt{u-mrg}$$

$$\texttt{l-mrg} \geq \textit{mfe}(u_i \in M, s_2) \geq \texttt{u-mrg}$$

Results and Discussion

`SIM` was primarily developed as a tool for designing synonymous mutations required for probing LRIs detected in viral genomes, such as previously performed by Gavazzi *et al.* in influenza A viruses *in vitro*[15]. With the help of `SIM`, experimentalists can simulate for the first time specific RNA-RNA interactions of viral coding- and non-coding sequence snippets, potentially forming stable long-range interactions. Using `SIM`, the search space of possible double-mutant sequences, forming a stable secondary structure with a comparable *mfe* to the wild-type structure and simultaneously disrupting the single-mutants structure, can be drastically reduced. Therefore, `SIM` can be used prior time- and cost-consuming wet-lab experiments to simulate promising double-mutant sequences, that preserve the wild-type LRI but are not functional as single-mutants. In this context, the use of `SIM` is not limited to RNA interactions between viral sequences, as the tool can be used for calculations between any two interacting RNA sequences. To validate *SIM* and to show that the tool is able to predict biological relevant mutants, we used two experimentally verified interactions, one in the influenza A virus H5N2 A/finch/England/2051/1991 strain and another in the hepatitis C virus type 1b strain (accession: AJ238799.1).

Application to influenza A virus

The influenza A virus (IAV) genome consists of 8 viral ribonucleoproteins (vRNPs) and each vRNP segment includes one of 8 different negative sense and single-stranded viral RNAs. It is hypothesized, that these segments are packed selectively through RNA-RNA interactions between the 8 segments[16,30–32]. To test this, Gavazzi *et al.*[15] conducted an *in vitro* mutation experiment in IAV, which is perfectly suited to validate our tool. They took a yet unconfirmed interaction from one of their previous experiments[14] between the PB1 and the NS segment and introduced four trans-complementary point substitutions by hand. An interaction between the PB1 mutant and NS mutant resulted in a similar viral titer than using the PB1 WT and NS WT segment. The introduction of only the PB1 mutant or NS mutant led in both cases to an attenuation of the viral replication.
To show that we are able to obtain the same experimental results as Gavazzi *et al.*[15] computationally, we adjusted some key parameters of `SIM`. Importantly, we limited the number of possible mutation pairs to four and calculated the *mfe* between the WT and mutant (mut) combinations in mere seconds. We further increased the search space of single mutants through setting the filter percentage parameter to 0.6, derived from the difference between the lowest single-mutant and the WT *mfe* value ($-24.60\ \frac{kcal}{mol}$ / $-14.00\ \frac{kcal}{mol}$ = 0.569; see **Fig. 2 a and c**). Because there

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

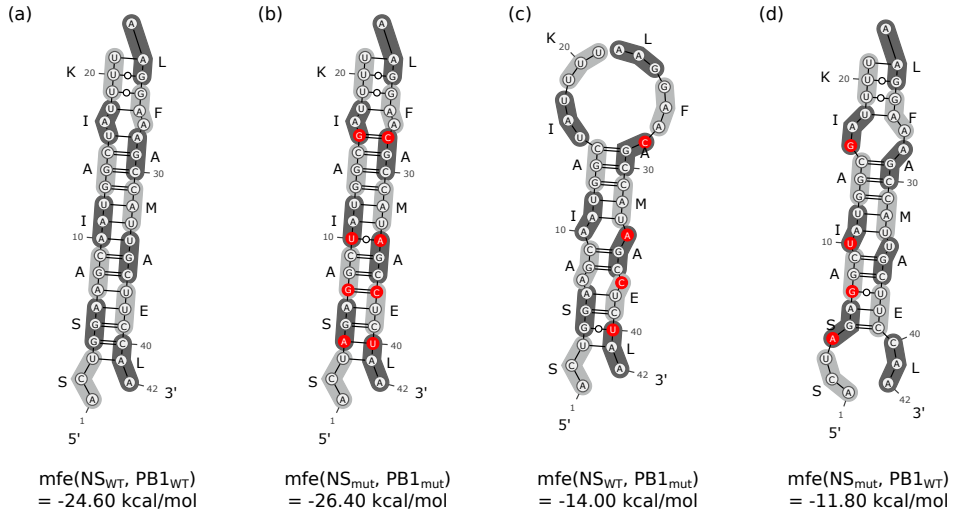

Figure 2: IAV interactions like previously shown by Gavazzi *et al.* We adjusted the parameters of `SIM` and allowed only four mutation pairs in maximum to calculate the *mfe* values for mutated sequence combinations of the NS and PB1 segments like previously shown by Gavazzi *et al.*[15]. The *mfe* values and secondary structures are shown for **(a)** the wild-type interaction, **(b)** the double-mutant interaction, **(c)** the first single-mutant ($PB1_{mut}$) interaction, and **(d)** the second single-mutant (NS_{mut}) interaction. This visualization was automatically generated by `SIM` using `VARNA`.

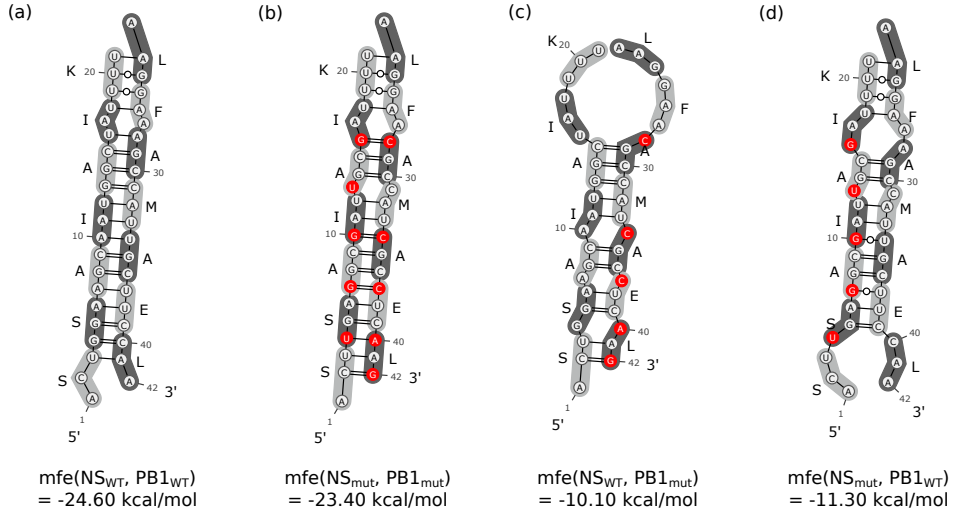

Figure 3: IAV interactions calculated by `SIM`. By applying `SIM` to the same IAV sequence snips of the NS and PB1 segments, we were able to show lower *mfe* differences between **(a)** the wild-type interaction and **(b)** the double-mutant interaction, as well as between the two single-mutant interactions **(c,d)**.

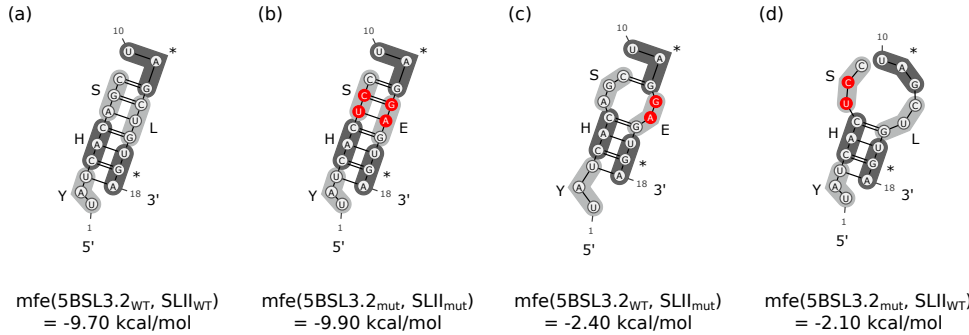

Figure 4: HCV interaction calculated with `SIM`. We used `SIM` to calculate a double-mutant interaction between a HCV protein-coding snip of 5BSL3.2 and an long-range interacting non-coding snip of the 3' SLII region. Again, our calculations show similar *mfe* values between the stable wild-type and double-mutant structures **(a,b)** as well as between the structure-destroying and less stable single-mutants **(c,d)**.

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

exist multiple double mutants with reasonable *mfe* values, we also had to define an exact search range for the double-mutant. The lower deviation was set to 1.07 and the upper deviation to 1.08, based on the difference between the double-mutant and WT *mfe* values (-26.40 $\frac{kcal}{mol}$ / -24.60 $\frac{kcal}{mol}$ = 1.0731; see **Fig. 2 a and b**). The SIM results of the secondary structure predictions and *mfe* values for the two IVA segment combinations are shown in **Fig. 2.** By only allowing a maximum of four mutation pairs, we were able to calculate the same structures and mutations as previously proposed by Gavazzi *et al.*[15]. Moreover, our results show a *mfe* difference of $1.80\ \frac{kcal}{mol}$ between wild-type and double-mutant (**Fig. 2 a and b**) and a *mfe* difference of $2.20\ \frac{kcal}{mol}$ between single-mutants (**Fig. 2 c and d**). Again, we want to point out that for the *in silico* experiment it is preferable to have a similar *mfe* between the two wild-type IAV segments *mfe*(NS_{WT}, $PB1_{WT}$) and the double-mutant *mfe*(NS_{mut}, $PB1_{mut}$) as well as between the single mutants *mfe*(NS_{WT}, $PB1_{mut}$) and *mfe*(NS_{mut}, $PB1_{WT}$).

In a next step and by using SIM with default parameters, we were able to calculate a double-mutant that not only reflects the results of Gavazzi *et al.*[15], but also shows slightly lower *mfe* differences between the wild-type and the double-mutant ($1.20\ \frac{kcal}{mol}$) as well as between the two single-mutants ($1.20\ \frac{kcal}{mol}$), see **Fig. 3**.

Therefore, the interaction strengths of *mfe*(NS_{WT}, $PB1_{WT}$) and *mfe*(NS_{mut}, $PB1_{mut}$), as well as *mfe*(NS_{WT}, $PB1_{mut}$) and *mfe*(NS_{mut}, $PB1_{WT}$) are more closely to each other in comparison to the results of Gavazzi *et al.*[15]. As a possible disadvantage, our simulation had to introduce another point mutation into each single-stranded RNA snip (**Fig. 3**).

Application to hepatitis C virus

The hepatitis C virus (HCV) genome consists of a positive single-stranded RNA of about 10 kb in length. This RNA is translated into a single polyprotein that is later cleaved into four structural (C, E1, E2, p7) and six nonstructural (NS2, NS3, NS4A, NS4B, NS5A, RdRp) proteins by viral and host proteases[33,34]. Both UTR regions of the viral genome are highly structured and have been extensively studied in the past[33,35–38].

To validate our tool, we have chosen a well-studied interaction[8] between the 3' UTR and the end region of the ORF encoding the HCV polyprotein. Several studies[9,33,39–42] have already shown that the RNA replication highly depends on the conserved structures of the X-tail[33,43] sequence contained in the 3' UTR. This sequence is presumed to contain three experimentally verified stem-loops (SLI, SLII, SLIII)[44] which may interact with other segments of the viral genome through long-range RNA-RNA interactions to regulate replication[40]. The LRI between the free nucleotides of the hairpin from the 3' SLII structure and the free nucleotides of the hairpin from the 5BSL3.2 structure have been investigated in many HCV studies.

The interaction was first verified by Friebe *et al.*[8] and was also computationally predicted with LRIscan by Fricke *et al.*[5]. Furthermore, we have chosen this interaction as an example, since on interacting segment is located in a coding region and the other segment in a non-coding region. Applying SIM on this interaction resulted in two compensatory point mutations in each snip (**Fig. 4**). As the 3' SLII region is non-coding, only the point mutations in the snip from the 5BSL3.2 structure are silent. Our results show, that the wild-type structure *mfe*($5BSL3.2_{WT}$, $SLII_{WT}$) should have a similar strength compared to the calculated double-mutant structure *mfe*($5BSL3.2_{mut}$, $SLII_{mut}$), see **Fig. 4**. Additionally, both *mfe*($5BSL3.2_{WT}$, $SLII_{mut}$) and *mfe*($5BSL3.2_{mut}$, $SLII_{WT}$) weaken the interaction significantly.

We suggest, that our simulated mutations in HCV may be used to verify the given long-range interaction. Taking all the new long-range interactions found by Fricke *et al.*[5] into account, we propose that our tool can be used to create mutation experiments for every predicted LRI to provide evidence for a biological function of these interactions. Such experiments would be especially interesting for IAV, where the exact packaging process of the vRNP segments is not yet fully understood.

Conclusion

With the development of SIM, we offer a simple and fast way to analyze possible interactions between vRNPs and other RNA molecules. The tool can be used to heavily reduce the search space of possible synonymous mutation interactions between two RNAs. Another difficulty when creating silent IAV mutations lies in preserving the codons on the positive strand, while mutating the negative strand, and is also intercepted by SIM. Furthermore, our tool provides a significant speedup, not only in the verification of interactions in these two viruses, but also for many other virus families. Our simulations will help to gather a deeper understanding of the translation and replication processes in viruses and also how long-range interactions are regulating these.

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

A promising approach for the future would be the combined use of `LRIscan` and `SIM` to detect previously unknown LRIs and at the same time perform possible mutational verification experiments for each LRI.

Author Contributions DD performed the design, development and programming of the tool and wrote the main draft of the paper. MH, BI, and MM contributed in writing, discussions and proofreading of the final manuscript. All authors read and approved the final manuscript.

References

[1] F. Hufsky, B. Ibrahim, M. Beer, L. Deng, P. L. Mercier, D. P. McMahon, M. Palmarini, V. Thiel, and M. Marz. Virologists-heroes need weapons. *PLOS Pathog*, 14: e1006771, 2018. ISSN 1553-7374.

[2] B. Ibrahim, D. P. McMahon, F. Hufsky, M. Beer, L. Deng, P. L. Mercier, M. Palmarini, V. Thiel, and M. Marz. A new era of virus bioinformatics. *Virus Res*, 251:86–90, 2018. ISSN 1872-7492.

[3] M. Hölzer and M. Marz. Software Dedicated to Virus Sequence Analysis "Bioinformatics Goes Viral". *Adv Virus Res*, 2017.

[4] B. L. Nicholson and K. A. White. Functional long-range RNA-RNA interactions in positive-strand RNA viruses. *Nature reviews. Microbiology*, 12:493–504, July 2014. ISSN 1740-1534.

[5] M. Fricke, N. Dünnes, M. Zayas, R. Bartenschlager, M. Niepmann, and M. Marz. Conserved RNA secondary structures and long-range interactions in hepatitis C viruses. *RNA*, 21(7):1219–1232, 2015.

[6] C. Romero-López and A. Berzal-Herranz. The functional RNA domain 5BSL3.2 within the NS5B coding sequence influences hepatitis C virus IRES-mediated translation. *Cellular and molecular life sciences : CMLS*, 69:103–113, January 2012. ISSN 1420-9071.

[7] A. Tuplin, M. Struthers, P. Simmonds, and D. J. Evans. A twist in the tail: SHAPE mapping of long-range interactions and structural rearrangements of RNA elements involved in HCV replication. *Nucleic acids research*, 40: 6908–6921, August 2012. ISSN 1362-4962.

[8] P. Friebe, J. Boudet, J.-P. Simorre, and R. Bartenschlager. Kissing-loop interaction in the 3' end of the hepatitis C virus genome essential for RNA replication. *Journal of virology*, 79:380–392, January 2005. ISSN 0022-538X.

[9] Y. Liu, E. Wimmer, and A. V. Paul. Cis-acting RNA elements in human and animal plus-strand RNA viruses. *Biochim Biophys Acta*, 1789(9):495–517, 2009.

[10] M. Honda, R. Rijnbrand, G. Abell, D. Kim, and S. M. Lemon. Natural variation in translational activities of the 5' nontranslated RNAs of hepatitis C virus genotypes 1a and 1b: evidence for a long-range RNA-RNA interaction outside of the internal ribosomal entry site. *Journal of virology*, 73:4941–4951, June 1999. ISSN 0022-538X.

[11] T. H. Wang, R. C. Rijnbrand, and S. M. Lemon. Core protein-coding sequence, but not core protein, modulates the efficiency of cap-independent translation directed by the internal ribosome entry site of hepatitis C virus. *Journal of virology*, 74:11347–11358, December 2000. ISSN 0022-538X.

[12] Y. K. Kim, S. H. Lee, C. S. Kim, S. K. Seol, and S. K. Jang. Long-range RNA-RNA interaction between the 5' nontranslated region and the core-coding sequences of hepatitis C virus modulates the IRES-dependent translation. *RNA (New York, N.Y.)*, 9:599–606, May 2003. ISSN 1355-8382.

[13] N. Beguiristain, H. D. Robertson, and J. Gómez. RNase III cleavage demonstrates a long range RNA: RNA duplex element flanking the hepatitis C virus internal ribosome entry site. *Nucleic acids research*, 33:5250–5261, 2005. ISSN 1362-4962.

[14] C. Gavazzi, C. Isel, E. Fournier, V. Moules, A. Cavalier, D. Thomas, B. Lina, and R. Marquet. An in vitro network of intermolecular interactions between viral RNA segments of an avian H5N2 influenza A virus: comparison with a human H3N2 virus. *Nucleic Acids Res*, 41:1241–1254, 2013. ISSN 1362-4962.

[15] C. Gavazzi, M. Yver, C. Isel, R. P. Smyth, M. Rosa-Calatrava, B. Lina, V. Moulès, and R. Marquet. A functional sequence-specific interaction between influenza A virus genomic RNA segments. *Proc Natl Acad Sci USA*, 110:16604–16609, 2013. ISSN 1091-6490.

[16] E. Fournier, V. Moules, B. Essere, J.-C. Paillart, J.-D. Sirbat, C. Isel, A. Cavalier, J.-P. Rolland, D. Thomas, B. Lina, and R. Marquet. A supramolecular assembly formed by influenza A virus genomic RNA segments. *Nucleic Acids Res*, 40:2197–2209, 2012. ISSN 1362-4962.

[17] E. Fournier, V. Moules, B. Essere, J.-C. Paillart, J.-D. Sirbat, A. Cavalier, J.-P. Rolland, D. Thomas, B. Lina, C. Isel, and R. Marquet. Interaction network linking the human H3N2 influenza A virus genomic RNA segments. *Vaccine*, 30:7359–7367, 2012. ISSN 1873-2518.

[18] C. Romero-López and A. Berzal-Herranz. A long-range RNA-RNA interaction between the 5' and 3' ends of the HCV genome. *RNA (New York, N.Y.)*, 15:1740–1752, September 2009. ISSN 1469-9001.

[19] E. Lam, A. Kam, and J. Waldispühl. corRna: a web server for predicting multiple-point deleterious mutations in structural RNAs. *Nucleic acids research*, 39:W160–W166, July 2011. ISSN 1362-4962.

[20] R. Sabarinathan, H. Tafer, S. E. Seemann, I. L. Hofacker, P. F. Stadler, and J. Gorodkin. The RNAsnp web server: predicting SNP effects on local RNA secondary structure. *Nucleic acids research*, 41:W475–W479, July 2013. ISSN 1362-4962.

[21] A. Churkin, I. Gabdank, and D. Barash. The RNAmute web server for the mutational analysis of RNA secondary structures. *Nucleic acids research*, 39:W92–W99, July 2011. ISSN 1362-4962.

[22] W. Shu, X. Bo, R. Liu, D. Zhao, Z. Zheng, and S. Wang. RDMAS: a web server for RNA deleterious mutation analysis. *BMC bioinformatics*, 7:404, September 2006. ISSN 1471-2105.

[23] J. Waldispühl, S. Devadas, B. Berger, and P. Clote. RNAmutants: a web server to explore the mutational landscape of RNA secondary structures. *Nucleic acids research*, 37:W281–W286, July 2009. ISSN 1362-4962.

[24] D. Barash and A. Churkin. Mutational analysis in RNAs: comparing programs for RNA deleterious mutation prediction. *Briefings in bioinformatics*, 12:104–114, March 2011. ISSN 1477-4054.

bioRxiv preprint doi: https://doi.org/10.1101/424002; this version posted November 1, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[25] M. Hamada, Y. Ono, H. Kiryu, K. Sato, Y. Kato, T. Fukunaga, R. Mori, and K. Asai. Rtools: a web server for various secondary structural analyses on single RNA sequences. *Nucleic acids research*, 44:W302–W307, July 2016. ISSN 1362-4962.

[26] J. Waldispühl, S. Devadas, B. Berger, and P. Clote. Efficient algorithms for probing the RNA mutation landscape. *PLoS computational biology*, 4:e1000124, August 2008. ISSN 1553-7358.

[27] R. Lorenz, S. H. Bernhart, C. Höner Zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker. ViennaRNA package 2.0. *Algorithms Mol Biol*, 6:26, 2011. ISSN 1748-7188.

[28] K. Darty, A. Denise, and Y. Ponty. VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics (Oxford, England)*, 25:1974–1975, August 2009. ISSN 1367-4811.

[29] B. Harrington. Inkscape. http://www.inkscape.org/, 2004.

[30] E. C. Hutchinson, J. C. von Kirchbach, J. R. Gog, and P. Digard. Genome packaging in influenza A virus. *J Gen Virol*, 91:313–328, 2010. ISSN 1465-2099.

[31] T. Noda, H. Sagara, A. Yen, A. Takada, H. Kida, R. H. Cheng, and Y. Kawaoka. Architecture of ribonucleoprotein complexes in influenza A virus particles. *Nature*, 439: 490–492, 2006. ISSN 1476-4687.

[32] Y. Fujii, H. Goto, T. Watanabe, T. Yoshida, and Y. Kawaoka. Selective incorporation of influenza virus RNA segments into virions. *Proc Natl Acad Sci USA*, 100: 2002–2007, 2003. ISSN 0027-8424.

[33] V. Lohmann. Hepatitis C virus RNA replication. *Curr Top Microbiol Immunol*, 369:167–198, 2013. ISSN 0070-217X.

[34] D. Moradpour and F. Penin. Hepatitis C virus proteins: from structure to function. *Current topics in microbiology and immunology*, 369:113–142, 2013. ISSN 0070-217X.

[35] E. A. Brown, H. Zhang, L. H. Ping, and S. M. Lemon. Secondary structure of the 5' nontranslated regions of hepatitis C virus and pestivirus genomic RNAs. *Nucleic acids research*, 20:5041–5045, October 1992. ISSN 0305-1048.

[36] M. Honda, M. R. Beard, L. H. Ping, and S. M. Lemon. A phylogenetically conserved stem-loop structure at the 5' border of the internal ribosome entry site of hepatitis C virus is required for cap-independent viral translation. *Journal of virology*, 73:1165–1174, February 1999. ISSN 0022-538X.

[37] P. Friebe and R. Bartenschlager. Role of RNA structures in genome terminal sequences of the hepatitis C virus for replication and assembly. *Journal of virology*, 83:11989–11995, November 2009. ISSN 1098-5514.

[38] M. Niepmann. Hepatitis C virus RNA translation. *Current topics in microbiology and immunology*, 369:143–166, 2013. ISSN 0070-217X.

[39] M. Yanagi, M. St Claire, S. U. Emerson, R. H. Purcell, and J. Bukh. *In vivo* analysis of the 3' untranslated region of the hepatitis C virus after *in vitro* mutagenesis of an infectious cDNA clone. *Proceedings of the National Academy of Sciences of the United States of America*, 96: 2291–2295, March 1999. ISSN 0027-8424.

[40] P. Friebe and R. Bartenschlager. Genetic analysis of sequences in the 3' nontranslated region of hepatitis C virus that are important for RNA replication. *Journal of virology*, 76:5326–5338, June 2002. ISSN 0022-538X.

[41] M. Yi and S. M. Lemon. 3' nontranslated RNA signals required for replication of hepatitis C virus RNA. *Journal of virology*, 77:3557–3568, March 2003. ISSN 0022-538X.

[42] S. You and C. M. Rice. 3' RNA elements in hepatitis C virus replication: kissing partners and long poly(U). *Journal of virology*, 82:184–195, January 2008. ISSN 1098-5514.

[43] T. Tanaka, N. Kato, M. J. Cho, and K. Shimotohno. A novel sequence found at the 3' terminus of hepatitis C virus genome. *Biochemical and biophysical research communications*, 215:744–749, October 1995. ISSN 0006-291X.

[44] K. J. Blight and C. M. Rice. Secondary structure determination of the conserved 98-base sequence at the 3' terminus of hepatitis C virus genome RNA. *Journal of virology*, 71:7345–7352, October 1997. ISSN 0022-538X.