ViR: a tool to account for intrasample variability in the detection of viral integrations

Elisa Pischedda¹, Cristina Crava^{1,2}, Martina Carlassara¹, Leila Gasmi¹, Mariangela Bonizzoni^{1,*}

¹ Department of Biology and Biotechnology, University of Pavia, Pavia

² ERI BIOTECMED, Universitat de Valencia, Valencia

*corresponding author

ABSTRACT

Lateral gene transfer (LT) from viruses to eukaryotic cells is a well-recognized phenomenon. Somatic integrations of viruses have been linked to persistent viral infection and genotoxic effects, including various types of cancer. As a consequence, several bioinformatic tools have been developed to identify viral sequences integrated into the human genome. Viral sequences that integrate into germline cells can be transmitted vertically, be maintained in host genomes and be co-opted for host functions. Endogenous viral elements (EVEs) have long been known, but the extent of their widespread occurrence has only been recently appreciated. Modern genomic sequencing analyses showed that eukaryotic genomes may harbor hundreds of EVEs, which derive not only from DNA viruses and retroviruses, but also from nonretroviral RNA viruses and are mostly enriched in repetitive regions of the genome. Despite being increasingly recognized as important players in different biological processes such as regulation of expression and immunity, the study of EVEs in non-model organisms has rarely gone beyond their characterization from annotated reference genomes because of the lack of computational methods suited to solve signals for EVEs in repetitive DNA. To fill this gap, we developed ViR, a pipeline which ameliorates the detection of integration sites by solving the dispersion of reads in genome assemblies that are rich of repetitive DNA. Using paired-end whole genome sequencing (WGS) data and a user-built database of viral genomes, ViR selects the best candidate couples of reads supporting an integration site by solving the dispersion of reads resulting from intrasample variability. We benchmarked ViR to work with sequencing data from both single and pooled DNA samples and show its applicability using WGS data of a non-model organism, the arboviral vector *Aedes albopictus*. Viral integrations predicted by ViR were molecularly validated supporting the accuracy of ViR results. Additionally, ViR can be readily adopted to detect any LT event providing *ad hoc* non-host sequences to interrogate.

INTRODUCTION

The transfer of genetic material between separate evolutionary lineages is a recognized event that occurs not only among prokaryotes, but also between viruses and eukaryotic cells (1). Somatic integrations of different viral species, among the best of known of which are the human papilloma virus, hepatitis B and C viruses and the Epstein-Barr virus, have been linked to genotoxic effects possibly progressing into cancer (2). Consequently, several pipelines have been developed to identify viral sequences integrated into the human genome using whole-genome sequencing (WGS) data (i.e. HIVID, SummonChimera, Vy-PER; HGT-ID, ViFi, VirTect, BS-virus-finder) (3–9). Each of these computational methods is differentially versatile in terms of data input format (i.e. RNA-seq, DNA-seq or bisulfite sequencing data) reference viral databases or customization opportunities, sensitivity and accuracy and CPU requirements, as previously reviewed (10). All of these pipelines have in common having been designed for the well-annotated human genome. Viruses can also integrate into the genome of germline cells. The persistence and the outcome of these integrations, which are vertically transmitted, depend on their effects on the fitness of the host. If deleterious, integrations are lost. If they occur in chromosomal locations that are not transcribed or lack regulative functions, integrations may persist and accumulate mutations. There are also examples of viral sequences being co-opted by the host and exerting beneficial functions (11). The existence of these Endogenous Viral Elements (EVEs) has long been known, with studies focusing mainly on EVEs from retroviruses in mammalian genomes (12, 13). The recent development of modern genomic sequencing approaches has opened to the study of non-model organisms. The genomes of organisms as different as arthropods (i.e. ticks, hematophagous and non-hematophagous insects), fish (i.e. zebrafish), snakes, birds, vertebrates (i.e. primates, mouse, rat, opossum) and plants were shown to host EVEs, which derive not only from DNA viruses and retroviruses, but also from nonretroviral RNA viruses (14–22). In these non-model organisms, EVEs range widely in numbers, for instance hundreds or less than 10 EVEs were identified in the genomes of *Aedes* or *Anopheles* mosquitoes, respectively (20, 23). Additionally EVEs of non-model organisms tend to occur in repetitive DNA, mostly in association with transposable element (TE) sequences and in Piwi-interacting RNA (piRNA) clusters (20, 23, 24). These newly-identified EVEs have been increasingly recognized as important players in different biological processes such as antiviral immunity and regulation of expression (25, 26). However, rarely studies of EVEs in non-model organisms have gone beyond their characterization from annotated reference genomes. The lack of bioinformatic tools able to account for repetitive DNA when mapping WGS data to a reference genome is hindering our ability to detect integration sites different than those already annotated in the reference genome, thus to understand the widespread occurrence and polymorphism of EVEs in host genomes and testing hypothesis on EVE function. To

ameliorate this issue, we have developed ViR, a new bioinformatics pipeline which improves the characterization of integration sites by solving the dispersion of reads in genome sequences that are rich of repetitive DNA. We benchmarked ViR with low- and high-coverage WGS data from *Aedes albopictus*, to date the mosquito species with the largest genome size and highest TE content among Culicinae (27, 28). We extensively evaluated ViR performance using WGS data from both single and pooled DNA samples, followed by molecular validation of predicted insertion sites. Importantly, ViR can be adopted to interrogate any genome for the presence of non-host sequences, which will facilitate studies of lateral gene transfer (LT).

MATERIALS AND METHODS

Pipeline implementation

The pipeline is composed of four scripts, which work in two modules. The first three scripts (ViR_RefineCandidates, ViR_SolveDispersion and ViR_AlignToGroup) work together to overcome the dispersion of reads due to intrasample variability (module 1). ViR_LTFinder is designed to run independently from the others, when testing for LT events of non-host sequences which have none or limited (defined by the user) sequence similarity to sequences of the host (module 2).

ViR_RefineCandidates. This script selects from a list of chimeric reads the best candidate pairs supporting a viral integration by filtering reads based on their sequence complexity, expressed as percentage of dinucleotides (default < 80%), imposing a minimum length recognized as viral (default 30 nucleotides) and removing mates that can align within a defined window in the reference genome (default 10000) based on Blast and BEDtools packages, respectively (29, 30) (Fig. 1A).

Inputs for ViR_RefineCandidates are a text file of paired-end chimeric reads and the SAM file of the reads aligned to the host genome. Paired-end chimeric reads are pairs in which one read maps to the host genome (hereafter called host read) and its pair maps to a virus (hereafter called viral read) (10). The script is versatile and accepts as input the text file listing chimeric pairs, independently of the used tools. If no pair of reads pass the filtering steps, the following is printed on terminal: ‘No viral reads pass filters. Pipeline STOP!’. Otherwise, an output file is generated, which, for each pair of reads, collects information on both the viral and host read. Information include read flags, sequence quality data and relative positions of the two mates. Details of the content of the output file are listed in Supplementary Table 1.

ViR_SolveDispersion. This script solves the dispersion of host reads by grouping together reads that map to regions of the genome with the same sequence (Fig. 1B). Reads that are grouped are called “read groups”; regions of the genome to which these groups of reads can equivalently map are called “equivalent regions”. Equivalent regions are regions of the host genome with the same sequence

because they contain the same repetitive element, or it is a sequence that has been erroneously assembled into different contigs or scaffolds. The script acquires as inputs a file listing all samples to analyze and the output directory of ViR_RefineCandidates. Reads mapping within equivalent regions are grouped together using the function “merge” from bedtools (Fig. 1B, step 1). Depending on the used blast settings, host reads may include also soft clipped reads (i.e. split reads in which a part of the read aligns to the host and the rest to a viral genome). The length of the equivalent region is defined by the user by setting the maximum distance among host reads and the minimum number of host reads within each region; defaults for these two parameters are 1000 base pairs (bp) and 2 reads, respectively. Then, identified read groups are compared in a pairwise mode in an iterative process in which read groups sharing more than a user-defined percentage of reads are collapsed in one; default percentage of reads shared between two read groups to be merged into one has been set to 80% (Fig. 1B, step 2). This procedure allows to identify the best candidate anchor genomic region of a candidate viral integration site (Fig. 1B, step 3). Each read group supports a candidate viral integration. Within each candidate, all viral reads correspond to the same viral species. Options to describe the genomic context of each region are available by providing adequate input files, such as the fasta file of the repeats of the host genome and/or the bed files of EVEs and piRNA clusters annotated in the reference genome.

ViR_AlignToGroup. This script predicts the right and left sides of the integration site by realigning reads supporting each candidate viral integration against the sequence of the equivalent region. First, for each candidate, this script extracts the host reads with their viral pair from the SAM file of the reads analyzed by ViR_RefineCandidates using the command-line utility “grep” (<https://www.gnu.org/software/grep/manual/grep.html>); the SAM file is converted into a BAM file using the function “view” of SAMtools (31); the BAM file is converted into a FASTQ format using the function “bamtofastq” from BEDtools (29) (Fig. 1C, step 1). Then, the script obtains the sequence of the equivalent region in fasta using the BEDtools function “getfasta” (29) (Fig. 1C, step 2). Reads from step 1 are re-aligned to the sequence of the equivalent region using “bwa mem” with default parameters (32). By taking advantage of the flags of alignment of each read of all chimeric pairs and eventual soft clipped reads, the left and right sides of the integration point can be predicted using Trinity (Fig. 1C, step 3). Even if no assemblies are created, flags of alignment are used to predict the direction of the integration sites (<https://broadinstitute.github.io/picard/explain-flags.html>) (Fig. 1C, step 4).

ViR_LTFinder. This script is designed to test for an integration from non-host sequences which have no, or a user-defined percentage of, similarity to host sequences. WGS reads are mapped to a selected non-host sequence (i.e. an entire genome or selected portions) using “bwa mem” with default

parameters (32). Mapped reads are extracted using the function “view” of SAMtools (31) (Fig. 1D, step 1). These reads include pairs in which both mates map to the non-host sequence, chimeric reads and soft-clipped reads. The aligned reads are converted into FASTQ format using the function “bamtofastq” from BEDtools (29) (Fig. 1D, step 2) and used for *de-novo* local assembly using Trinity (33) (Fig. 1D, step 3). A consensus sequence is built if any instances of LT are identified. Output of ViR_LTFinder include files for visualization of the aligned reads using the Integrated Genomics Viewer (IGV) tool (34).

Estimating the gain in solving read dispersion

The utility of ViR in solving read dispersion was assessed using the concept of the ‘Gain Index’ parameter from the Information Theory (35). This index estimates the weight that each attribute has in building a classification of entities, given n entities each defined through various attributes (35). The maximization of the Gain index reflects the power of an attribute in segregating entities to different classes. In our case, we used this index to evaluate the gain of enclosing in a single ‘equivalent region’, reads that had been originally assigned to different loci, our classes. The attribute is the “equivalent region” identified by ViR_SolveDispersion and the entities are all the reads supporting candidate integrations (i.e. output of ViR_RefineCandidates).

The gain of running ViR is estimated through the formula:

$$\text{Normalised Dispersion Gain} = \frac{I - I_{res}}{I}$$

Where I is the entropy.

Entropy is the dispersion of the reads supporting each locus ID in a sample and it is quantified by the formula:

$$I = - \sum_l p(l) * \log_e p(l)$$

where l is the locus ID, meaning the host genomic coordinates of the candidate integration identified before ViR. $p(l)$ is the relative frequency of the reads assigned to the locus ID l . Entropy is 0 when only one locus ID (i.e. one candidate integration site) is identified in the sample (i.e. WGS dataset). Entropy is > 0 , when more than one locus ID is identified in the sample.

And I_{res} is the residual information.

The residual information is defined by the formula:

$$I_{res} = - \sum_g p(g) \sum_l p(l|g) * \log_e p(l|g)$$

Where g is the ID of the equivalent region identified by ViR_SolveDispersion, $p(g)$ is the relative frequency of the reads in the equivalent region g and $p(l|g)$ is the relative frequency of reads assigned to the locus ID l in the equivalent region g .

The difference between I and I_{res} is evaluated for each sample. The ratio of the difference with respect to the initial entropy is calculated to normalize across samples, which were obtained using different experimental set ups (i.e. WGS from single or pools). This operation results in a *Normalised Dispersion Gain*, which ranges between 0 and 1. The closer the value of *Normalised Dispersion Gain* is to 0, the higher is the gain of ViR in solving the dispersion of reads. To favor intuitive interpretation of results, we show Solve Dispersion Gain as:

$$Solve\ Dispersion\ Gain = 1 - Normalised\ Dispersion\ Gain = 1 - \frac{I - I_{res}}{I} = \frac{I_{res}}{I}$$

Values of *Solve Dispersion Gain* > 0 are found when ViR was able to identify a unique equivalent region for at least two different reads previously assigned to two different loci ID. The higher the value of *Solve Dispersion Gain*, the higher is the performance of ViR.

Molecular validation of ViR-predicted integration sites

We run ViR on WGS data of *Ae. albopictus* and identified a total of seven candidate viral integrations absent in the reference genome assemblies, AloF1 and AalbF2 (27, 28). PCR primers were designed on the basis of these candidates (Table 1) and used on DNA extracted from different mosquitoes than those used as source of WGS data. *Aedes albopictus* mosquitoes are reared at the insectary of the University of Pavia as previously described (23). Genomic DNA was extracted from individual mosquitoes using the Promega Wizard® Genomic DNA Purification Kit, according to manufacturer's protocol. PCR reactions were carried out with the DreamTaq Green PCR Master Mix (ThermoFisher) using 1 µl of genomic DNA. Amplified bands were purified with ExoSAP-IT kit (ThermoFisher) and send to Macrogen (Madrid, Spain) for Sanger sequencing. Sequences were analyzed with Bioedit (36).

RESULTS

Overview of the ViR pipeline

ViR is designed to solve the dispersion of reads due to an integration site occurring in repetitive or misassembled regions of a genome. Any of the currently available tools to identify viral integrations from paired-end sequencing data operates by selecting chimeric pairs, in which one read maps to the host genome (the host read) and the other to the viral genome (the viral read) (10). If the integration site occurs in a repeated region or in a genomic region that has been assembled into different contigs or scaffolds, host reads may be distributed across these “equivalent” genomic positions, and, as a consequence, the signal for the integration site, expressed in terms of host reads coverage, may not reach threshold of detection. To ameliorate this, we developed ViR, a pipeline composed of four scripts divided into two modules (Fig. 1).

Using the script `ViR_SolveDispersion`, reads that map to equivalent regions of the genome are grouped together (Fig. 1B). These groups represent anchor genomic regions where candidate viral integrations occurred. Additional information on these regions can be collected by providing input files with custom data (i.e. mapping coordinates of EVEs annotated in the reference genome, of piRNA clusters, of TEs, etc.). This additional information is useful to distinguish between polymorphisms of EVEs *versus* new integrations from the same virus. The best candidate pair of reads supporting a viral integration are identified by running the script `ViR_RefineCandidates` (Fig. 1A), which parsed viral reads through different filtering steps. This procedure should be performed before running the script `ViR_SolveDispersion` to reduce the computational time to solve the dispersion of host reads. Following `ViR_SolveDispersion`, the sequence of each equivalent genomic region is extracted, and host reads are re-aligned to it. This procedure allows to predict the left and right sides of the integration site through the use of alignment flags (Fig. 1C).

`ViR_RefineCandidates`, `ViR_SolveDispersion` and `Vir_AlignToGroup` work together to ameliorate the detection of integration sites by solving the dispersion of reads due to intrasample variability (module 1).

The pipeline comprises an additional script, `ViR_LTFinder`, which runs independently from the others (module 2). This script uses WGS data to map against a non-host sequence (i.e. entire genomes, genes, transposable elements). Aligned reads are extracted and used for *de novo* local alignment using Trinity building a consensus sequence for the LT event (Fig. 1D). Eventual similarities between the tested non-host sequence and sequences of the host may lead to unambiguities, for instance if searching for integrations from a viral species for which EVEs are already annotated in the host genome, thus users have to decide a threshold for the percentage of similarity between their non-host sequence and eventual host sequences to avoid misinterpretation of results from `ViR_LTFinder`.

Benchmarking ViR

ViR was run using WGS data from the arboviral vector *Ae. albopictus*. *Aedes albopictus* has the largest mosquito genome sequenced to date, the majority of which is repetitive DNA (27, 28). For *Ae. albopictus*, three genome assemblies are available, which differ greatly in the number of scaffolds and are all larger than the expected size of the genome based on cytofluorimetric estimates, suggesting duplications (27, 28, 37). We selected to run ViR using two assemblies, AaloF1 and AalbF2, which have 154782 and 2197 scaffolds, respectively. Hundreds of EVEs from various taxonomic viral categories are annotated in AaloF1 and AalbF2 (23, 28), giving further complexity.

We generated WGS data from DNA of both single and pool DNA samples and at different coverage to compare the accuracy of ViR across different conditions (Fig. 2A). Single samples consisted in twenty-two mosquitoes whose genomes were analyzed independently. WGS libraries were generated and sequenced on an Illumina HiSeqX platform as previously described (38). Single samples were sequenced at an approximate coverage of 30x. Pool samples were 6 samples consisting each of the DNA from 30 mosquitoes. Three samples were sequenced at an approximate coverage of 30x and called pool30; the remaining samples were sequenced at an approximate coverage of 60x and called pool60. A list of chimeric pairs, indicative of candidate integration sites, were obtained for each sample running Vy-PER (3) with a custom-made database of viral genomes (Supplementary Table 1). Using 4 CPUs and 16 Gb RAM per sample, VyPER produced results in between 5-33 hours in AaloF1 and 6-68 hours in AalbF2 (Table 2). Chimeric reads identified by VyPER are spread across different regions of the host genomes (Fig. 2A) and viral reads include homopolymers or low-complexity sequences.

Testing module 1.

ViR_RefineCandidates and ViR_SolveDispersion were run for all samples using both AaloF1 and AalbF2 (Fig. 2A). Parameters set in ViR_RefineCandidates included filtering out viral reads with homopolymers (any combination of dinucleotides higher than 80%), viral matches shorter than 30 nucleotides and distance between read pairs higher than 10000 nucleotides, considering the maximum DNA fragment length produced in our WGS data. Parameters set in ViR_SolveDispersion included defining a read group when at least two host reads aligned within a window of 1000 nucleotides and merging read groups that shared more than 80% of reads. Publicly available annotations of EVEs, piRNA clusters and transposable elements were used to define the genomic context of each equivalent region (23, 28).

When ViR was run with AaloF1, five and two candidate viral integrations were identified in pool60 and pool30 samples, respectively, with a maximum number of reads per sample of 28. Viral integrations were also identified in eight out of the 22 tested single samples, with a maximum number of reads per sample of eight (Fig. 2A). When ViR was run with AalbF2, we identified a maximum of five viral integrations in pool60 samples and one in pool30 samples with a maximum number of reads per sample of 17. As for AaloF1, eight of the 22 single samples showed candidate viral integrations, with a maximum number of reads per sample of six (Fig. 2A).

Overall across all samples, a total of seven integrations, all of viruses from the Flaviviridae family were identified. Both the right and left sides of the integration were resolved bioinformatically for nrEVEnew-7. Only the left or right sides of the integration were bioinformatically predicted from nrEVEnew-1, nrEVEnew-2 and nrEVEnew-6 or nrEVEnew-4, nrEVEnew-5 and nrEVEnew-8, respectively. nrEVEnew-2, nrEVEnew-4, nrEVEnew-6 and nrEVEnew-7 were detected when running ViR using both AaloF1 and AalbF2; nrEVEnew-1, nrEVEnew-5 and nrEVEnew-8 were detected only when running ViR with AalbF2, possibly because of the higher completeness of AalbF2 (28) vs AaloF1 (27). Viral integrations identified running ViR in the genome of *Ae. albopictus* mosquitoes are described in Table 3.

Testing module 2.

Running module 1 of ViR led to the identification of both the right and left integration sides of nrEVEnew-7, but without the exact integration points (Fig. 2B). nrEVE-new-7 is a 187 bp sequence with similarity to Aedes Flavivirus (AeFV), with less than 78% nucleotide identity with EVEs already annotated in either AaloF1 (23) or AalbF2 (28). Thus, we implemented ViR-LTFinder to verify whether our WGS data contained reads encompassing the integrations site. Running ViR_LTFinder with our WGS data using as anchor the viral portion of nrEVEnew-7 built to a consensus sequence of 1237 bp including both integration sites (Fig. 2B, section a). ViR_LTFinder was further tested for nrEVEnew-1, nrEVEnew-2, nrEVEnew-4, nrEVEnew-5 and nrEVEnew-6. Results of ViR_LTFinder showed that nrEVEnew-4 and nrEVEnew-6 correspond respectively to the right and left regions of the same integration, for a total of 534 viral bp (Fig. 2B, section b). With ViR_LTFinder the viral sequence of nrEVEnew-2 (Supplementary Fig. 1, step a) was extended for 619 bp to the left predicting an integration break point and 636 bp to the right. The sequence on the right was still viral indicating nrEVE-new-2 is longer (Supplementary Fig. 1, step b). Thus, we repeated ViR_LTFinder as before. This operation produced two consensus sequences: the first one of 1474 bp including both the left and the right anchor sites on the mosquito genome and supporting an integration of 510 bp; the second

of 1987 bp extending the sequence obtained in the first step of 577 bp and supporting a longer integration (Supplementary Fig. 1, step c).

Molecular validation.

PCR-based molecular validation was obtained for all predicted viral integrations, with the exception of nrEVE-new8 and the left integration sites of nrEVE-new-2 (Fig. 2C). We cannot exclude this result is due to the rarity of these integrations. nrEVE-new8 was bioinformatically identified only in one Pool-60 sample. Unfortunately, we were unable to perform PCR on this pool because of the lack of DNA aliquots.

Testing ViR performance

We tested the benefit of running ViR by calculating the solve dispersion gain parameter. For each sample, we considered as dataset the whole list of reads resulting from ViR_RefineCandidates. We quantified the gain comparing the host-mapping loci identified by VyPER and the read groups created by ViR_SolveDispersion. As an example, in replicate 19 among the single samples, seven pairs of reads were identified by Vy-PER as candidate viral integrations. ViR solved the dispersion of these seven reads by grouping them into one group supporting nrEVEnew-4; two read remained ungrouped, resulting in a Solve Dispersion Gain value of 0,65 (Fig. 3A). In single samples, the median values of the solve dispersion gain were 0.47 and 0.5 when using AaloF1 and Aalb2, respectively. In pool samples, the median values in pool60 were 0.36 and 0.42 and in pool30 were 0.58 and 0.48 when using AaloF1 and Aalb2, respectively (Fig. 3B). Thus, we noticed a significant gain in using ViR, also in the improved version of the *Ae. albopictus* assembly, AalbF2 (28). Dispersion gain values were not different between single vs pools or between pools 30 vs pools 60 samples, indicating the gain is influenced by the sequencing strategy nor the depth of sequencing.

In each dataset a certain number of chimeric reads identified by Vy-PER could not be grouped. These reads have alignments in the genome distant from the others and cannot be used to predict candidate viral integrations. Even if they are not useful for the discovery of viral integrations, it is important to isolate them to avoid wasting time trying to interpret them, for this motif we included these ungrouped reads in the calculation of the Solve Dispersion Gain. These ungrouped reads were more abundant when testing ViR with AaloF1 than AaloF2 as a result of the higher fragmentation of the AaloF1 assembly (28).

DISCUSSION

The purpose of ViR is to provide reliable identification of EVEs when viral integrations occur in repetitive regions of host genomes or when using genome assemblies of non-model organisms which are still fragmented. Both these elements generate intra-host variability resulting in mis-alignments when trying to map short paired end reads. This issue is magnified in chimeric reads, which are the source of a candidate viral integration because in chimeric reads only one of the pairs maps to the host genome. ViR is not a new pipeline for the identification of EVEs, rather it works to ameliorate predictions. This is important because despite the large number of pipelines currently available to detect EVEs (for a review see (10)), all have been designed for well-annotated model genomes, *in primis* the human genome. The advent of next-generation sequencing technologies and metagenomic analyses facilitated the study of non-model organisms, the genomes of many of which were sequenced showing that LT from viruses may be as frequent in eukaryotes as in prokaryotes (17, 20). Additionally, integrated viral sequences in both plants and animals derive not only from DNA viruses and retroviruses but also from non-retroviral RNA viruses and are enriched in repetitive regions of the genome (20, 23–25, 28). To understand the widespread distribution of EVEs and their biological role, we need to characterize EVEs using WGS data from wild-collected samples or samples collected under hypothesis-driven experimental conditions. In the case of non-model organism, this pursuit requires being able to handle genomes rich in repetitive elements and/or fragmentation in the assembly. Towards this goal, we have developed ViR, a bioinformatic pipeline suited to account for intrasample variability in predicting candidate viral integrations. We show ViR performance by using WGS data from *Ae. albopictus*, the mosquito with the largest genome to date, a TE content >50% of the genome and two assemblies differing in completeness (27, 28). In the absence of a true set of viral integrations, ViR performance was tested calculating the dispersion gain, a parameter from the Information Theory (35) and molecularly validating predicted viral integrations. We demonstrated that ViR is able to solve the dispersion of reads supporting a viral integration similarly when using WGS data from single and pool samples and with a coverage of between 30 and 60x. We anticipate ViR will open new venues to explore the biology of EVEs, especially in non-model organisms. The design of ViR makes it compatible with any list of putative chimeric reads produced by any currently available EVE identification tool, giving users great flexibility. Importantly, while we generated ViR with the identification of EVEs in mind, its application can be extended to detect any LT event providing an *ad-hoc* sequence to interrogate.

AVAILABILITY

ViR is implemented in bash and python2.7 and is freely accessible on Github.

CONTACT

elisa.pischedda01@universitadipavia.it

SUPPLEMENTARY DATA

Supplementary data will be available in the online journal.

FUNDING

This work was supported by the European Research Council, ERC-CoG 682394 to M. Bonizzoni; the Italian Ministry of Education, University and Research, FARE-MIUR project R1623HZA5 to M. Bonizzoni and Dipartimenti Eccellenza Program 2018–2022 to Dept. of Biology and Biotechnology “L. Spallanzani”, University of Pavia; the Human Frontiers Science Foundation, Research Grant number RGP0007/2017 to M. Bonizzoni; *Conflict of interest statement*. Not declared.

REFERENCES

1. Keeling,P.J. and Palmer,J.D. (2008) Horizontal gene transfer in eukaryotic evolution. *Nat. Rev. Genet.*, **9**, 605–618.
2. Chen,Y., Williams,V., Filippova,M., Filippov,V. and Duerksen-Hughes,P. (2014) Viral Carcinogenesis: Factors Inducing DNA Damage and Virus Integration. *Cancers (Basel)*., **6**, 2155–2186.
3. Forster,M., Szymczak,S., Ellinghaus,D., Hemmrich,G., Rühlemann,M., Kraemer,L., Mucha,S., Wienbrandt,L., Stanulla,M. and Franke,A. (2015) Vy-PER: eliminating false positive detection of virus integration events in next generation sequencing data. *Sci. Rep.*, **5**, 11534.
4. Katz,J.P. and Pipas,J.M. (2014) SummonChimera infers integrated viral genomes with nucleotide precision from NGS data. *BMC Bioinformatics*, **15**, 348.
5. Nguyen,N.D., Deshpande,V., Luebeck,J., Mischel,P.S. and Bafna,V. (2018) ViFi: accurate detection of viral integration and mRNA fusion reveals indiscriminate and unregulated transcription in proximal genomic regions in cervical cancer. *Nucleic Acids Res.*, **46**, 3309–3325.
6. Xia,Y., Liu,Y., Deng,M. and Xi,R. (2019) Detecting virus integration sites based on multiple related sequencing data by VirTect. *BMC Med. Genomics*, **12**, 19.
7. Baheti,S., Tang,X., O’Brien,D.R., Chia,N., Roberts,L.R., Nelson,H., Boughey,J.C., Wang,L., Goetz,M.P., Kocher,J.-P.A., *et al.* (2018) HGT-ID: an efficient and sensitive workflow to

detect human-viral insertion sites using next-generation sequencing data. *BMC Bioinformatics*, **19**, 271.

8. Li,W., Zeng,X., Lee,N.P., Liu,X., Chen,S., Guo,B., Yi,S., Zhuang,X., Chen,F., Wang,G., *et al.* (2013) HIVID: An efficient method to detect HBV integration using low coverage sequencing. *Genomics*, **102**, 338–344.
9. Gao,S., Hu,X., Xu,F., Gao,C., Xiong,K., Zhao,X., Chen,H., Zhao,S., Wang,M., Fu,D., *et al.* (2018) BS-virus-finder: virus integration calling using bisulfite sequencing data. *Gigascience*, **7**.
10. Chen,X., Kost,J. and Li,D. (2019) Comprehensive comparative analysis of methods and software for identifying viral integrations. *Brief. Bioinform.*, **20**, 2088–2097.
11. Frank,J.A. and Feschotte,C. (2017) Co-option of endogenous viral sequences for host cell function. *Curr. Opin. Virol.*, **25**, 81–89.
12. Johnson,W.E. (2015) Endogenous Retroviruses in the Genomics Era. *Annu. Rev. Virol.*, **2**, 135–159.
13. Dewannieux,M. and Heidmann,T. (2013) Endogenous retroviruses: acquisition, amplification and taming of genome invaders. *Curr. Opin. Virol.*, **3**, 646–656.
14. Harper,G., Hull,R., Lockhart,B. and Olszewski,N. (2002) Viral sequences integrated into plant genomes. *Annu. Rev. Phytopathol.*, **40**, 119–136.
15. Belyi,V.A., Levine,A.J. and Skalka,A.M. (2010) Unexpected inheritance: multiple integrations of ancient bornavirus and ebolavirus/marburgvirus sequences in vertebrate Genomes. *PLoS Pathog.*, **6**, 1–13.
16. Horie,M., Honda,T., Suzuki,Y., Kobayashi,Y., Daito,T., Oshida,T., Ikuta,K., Jern,P., Gojobori,T., Coffin,J.M., *et al.* (2010) Endogenous non-retroviral RNA virus elements in mammalian genomes. *Nature*, **463**, 84–87.
17. Katzourakis,A. and Gifford,R.J. (2010) Endogenous viral elements in animal genomes. *PLoS Genet.*, **6**, e1001191.
18. Gilbert,C., Meik,J.M., Dashevsky,D., Card,D.C., Castoe,T.A. and Schaack,S. (2014) Endogenous hepadnaviruses, bornaviruses and circoviruses in snakes. *Proc. R. Soc. B Biol. Sci.*, **281**, 20141122.
19. Flynn,P.J. and Moreau,C.S. (2019) Assessing the Diversity of Endogenous Viruses Throughout Ant Genomes. *Front. Microbiol.*, **10**.
20. ter Horst,A.M., Nigg,J.C., Dekker,F.M. and Falk,B.W. (2019) Endogenous Viral Elements Are Widespread in Arthropod Genomes and Commonly Give Rise to PIWI-Interacting RNAs. *J. Virol.*, **93**, e02124-18.

21. Russo,A.G., Kelly,A.G., Enosi Tuipulotu,D., Tanaka,M.M. and White,P.A. (2019) Novel insights into endogenous RNA viral elements in *Ixodes scapularis* and other arbovirus vector genomes. *Virus Evol.*, **5**, 1–18.
22. Blair,C.D., Olson,K.E. and Bonizzoni,M. (2019) The Widespread Occurrence and Potential Biological Roles of Endogenous Viral Elements in Insect Genomes. *Curr. Issues Mol. Biol.*, 10.21775/cimb.034.013.
23. Palatini,U., Miesen,P., Carballar-Lejarazu,R., Ometto,L., Rizzo,E., Tu,Z., van Rij,R.P. and Bonizzoni,M. (2017) Comparative genomics shows that viral integrations are abundant and express piRNAs in the arboviral vectors *Aedes aegypti* and *Aedes albopictus*. *BMC Genomics*, **18**, 1–15.
24. Whitfield,Z.J., Dolan,P.T., Kunitomi,M., Tassetto,M., Seetin,M.G., Oh,S., Heiner,C., Paxinos,E. and Andino,R. (2017) The diversity, structure, and function of heritable adaptive immunity sequences in the *Aedes aegypti* genome. *Curr. Biol.*, **27**, 3511–3519.
25. Crava,C., Varghese,F.S., Pischedda,E., Halbach,R., Palatini,U., Marconcini,M., Mattia,A., Redmond,S., Afrane,Y., Ayala,D., *et al.* (2020) Immunity to infections in arboviral vectors by integrated viral sequences: an evolutionary perspective. *bioRxiv*, 10.1101/2020.04.02.022509.
26. Suzuki,Y., Baidaliuk,A., Miesen,P., Frangeul,L., Crist,A.B., Merklings,S.H., Fontaine,A., Lequime,S., Moltini-Conclois,I., Blanc,H., *et al.* (2020) Non-retroviral endogenous viral element limits cognate virus replication in *Aedes aegypti* ovaries. *bioRxiv*, 10.1101/2020.03.28.013441.
27. Chen,X.-G., Jiang,X., Gu,J., Xu,M., Wu,Y., Deng,Y., Zhang,C., Bonizzoni,M., Dermauw,W., Vontas,J., *et al.* (2015) Genome sequence of the Asian Tiger mosquito, *Aedes albopictus*, reveals insights into its biology, genetics and evolution. *Proc. Natl. Acad. Sci.*, **112**, E5907–E5915.
28. Palatini,U., Masri,R., Cosme,L., Koren,S., Thibaud-Nissen,F., Biedler,J., Krsticevic,F., Johnston,J., Halbach,R., Crawford,J., *et al.* (2020) Improved reference genome of the arboviral vector *Aedes albopictus*. *bioRxiv*, 10.1101/2020.02.28.969527.
29. Quinlan,A.R. and Hall,I.M. (2010) BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics*, **26**, 841–842.
30. Zhang,Z., Schwartz,S., Wagner,L. and Miller,W. (2000) A Greedy Algorithm for Aligning DNA Sequences. *J. Comput. Biol.*, **7**, 203–214.
31. Li,H., Handsaker,B., Wysoker,A., Fennell,T., Ruan,J., Homer,N., Marth,G., Abecasis,G. and Durbin,R. (2009) The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

32. Li,H. (2013) Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. **00**, 1–3.
33. Grabherr,M.G., Haas,B.J., Yassour,M., Levin,J.Z., Thompson,D.A., Amit,I., Adiconis,X., Fan,L., Raychowdhury,R., Zeng,Q., *et al.* (2011) Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat. Biotechnol.*, **29**, 644–652.
34. Robinson,J.T., Thorvaldsdóttir,H., Winckler,W., Guttman,M., Lander,E.S., Getz,G. and Mesirov,J.P. (2011) Integrative genomics viewer. *Nat. Biotechnol.*, **29**, 24–26.
35. Shannon,C.E. (1948) A Mathematical Theory of Communication. *Bell Syst. Tech. J.*, 10.1002/j.1538-7305.1948.tb01338.x.
36. Hall,T.A. (1999) BioEdit: a user-friendly biological sequence alignment editor and analysis program for Windows 95/98/NT. *Nucleic Acids Symp. Ser.*, **41**, 95–98.
37. Dritsou,V., Topalis,P., Windbichler,N., Simoni,A., Hall,A., Lawson,D., Hinsley,M., Hughes,D., Napolioni,V., Crucianelli,F., *et al.* (2015) A draft genome sequence of an invasive mosquito: an Italian *Aedes albopictus*. *Pathog. Glob. Health*, **109**, 207–220.
38. Marconcini,M., Hernandez,L., Iovino,G., Houé,V., Valerio,F., Palatini,U., Pischedda,E., Crawford,J.E., White,B.J., Lin,T., *et al.* (2019) Polymorphism analyses and protein modelling inform on functional specialization of Piwi clade genes in the arboviral vector *Aedes albopictus*. *PLoS Negl. Trop. Dis.*, **13**, e0007919.

Figure legends

Figure 1: Overview of ViR. ViR comprises four main scripts, organized into two modules (Mod1 and Mod2): ViR_RefineCandidates, ViR_SolveDispersion, ViR_AlignToGroup and LT_Finder. **(A)** ViR_RefineCandidates filters the viral mate of the pairs of reads to identify the best candidate viral reads. Inputs for the script are the SAM file of the alignment of WGS data and a text file with the paired-end chimeric reads. Chimeric reads are pairs in which one read maps to the host genome (hereafter called host read, in gray) and one to a viral sequence (hereafter called viral read, in red) (10). Viral reads are identified using on a user-built database. Flags of alignment and sequence quality information are extracted for the identified best candidates. **(B)** ViR_SolveDispersion is designed to identify groups of host reads supporting an integration site in equivalent genomic regions (step 1). Equivalent means all genomic regions to which grouped reads can align. Read groups are compared in a pair-wise mode to merge groups sharing a certain percentage of reads (step 2). Remaining read groups support candidate viral integrations (step 3). **(C)** For each identified read group ViR_AlignToGroup extracts the reads from the SAM file of the selected reads by ViR_RefineCandidates (step 1) and the sequence of the equivalent region of the group in FASTA format (step 2). Then, reads are re-aligned to the sequence of the equivalent region and flags of alignment are used to identify the left and right side of the integration site (step 3). Examples of flags supporting the right (i.e. 73, 133) and left (i.e. 117, 185) ends of the integration site are shown (step 4). **(D)** ViR_LTFinder is used when searching for viral integrations from viruses for which no EVEs are annotated in the reference genome. WGS raw reads (WGS FASTQ file) are aligned to a viral genome(s) (FASTA file) (step 1). Aligned reads are extracted, converted into FASTQ (step 2) and used for local *de-novo* assembly (step 3). Aligned reads include mates in which both reads map to the viral genome (in red), chimeric reads (viral read in red and host read in dashed grey) and mates in which one is a soft clipped read (viral portion in red, the rest in dashed gray). Mates are indicated by continuous thin line.

Figure 2: Benchmarking ViR. **(A)** Whole Genome Sequencing (WGS) data were generated from DNA of both single and pool DNA samples and at different coverage to compare the accuracy of ViR across different conditions. We selected to run ViR using two assemblies, AaloF1 and AalbF2 (28,29) with a custom database of viral genomes (Supplementary Table 2). A list of chimeric pairs, indicative of candidate integration sites, were obtained for each sample running Vy-PER (3). Module 1 of the ViR pipeline was run using the chimeric reads identified by Vy-PER to solve read dispersion, resulting in a total of seven candidate viral integrations across all samples. **(B)** **(a)** ViR_LTFinder was run with WGS data and the viral sequence of nrEVEnew-7 generating an extended consensus

(step 1). ViR_LTFinder was run a second time using this elongated viral sequence as input (step 2), which led to the identification of both integration sites (step 3); **(b)** ViR_LTFinder was run with WGS data and the viral sequences of nrEVEnew-4 and nrEVEnew-6 showing that they correspond respectively to the right and left regions of the same integration. **(C)** Molecular validation of the five candidate viral integrations. Each lane is the result of a PCR on mosquito genomic DNA with primers that were design to check the left integration site for nrEVE-new-2, and to check both the left and right integration points for the others viral integrations. '+' indicates the presence of the viral integration and '-' the absence.

Figure 3: Testing the gain of running ViR. **(A)** Seven pairs of reads were identified by Vy-PER as candidate viral integrations. ViR grouped five of these reads into a single group which led to the identification of nrEVEnew-4 and resulting in a Solve Dispersion Gain value of 0,65. **(B)** Values of Solve Dispersion Gain were calculated for all single and pool samples.

Table 1. List of PCR primers used to confirm ViR-predicted viral integrations from WGS data of *Aedes albopictus*.

nrEVE	Primer Sequence	F or R	Location
nrEVEnew-1	ATCTGCCACACCCGGGTT	F	<i>genome</i>
	GTTGTTCCCCGGGATAACGT	R	<i>genome</i>
nrEVEnew-2a	CCGCGCTCTCACTCAGTA	F	<i>genome</i>
	ACTGATGATCTTGTGCTGATGG	R	nrEVE
nrEVEnew-4/6	TTTTCACTCCCGGGACCTG	F	<i>genome</i>
	GTGTTGATCCTTTCCACCGT	R	<i>nrEVE</i>
nrEVEnew-5	GTATCCGAGGAACACAAGCG	F	nrEVE
	CTACACAACAGCCAACTCCG	R	<i>genome</i>
nrEVEnew-7	TGATTCGATTCTTACAACCCCTA	F	<i>genome</i>
	GCGTATAAGCGTGTTGACCT	R	nrEVE

Table 2. Per sample computational and time cost for VyPER implementation in different experimental conditions.

	Single sample		Pool30		Pool60	
	AaloF1	AalbF2	AaloF1	AalbF2	AaloF1	AalbF2
CPU	4	4	4	4	4	4
Memory	16 Gb	16 Gb	16 Gb	16 Gb	16 Gb	16 Gb
Mean Time	8 h	11 h	5 h	6 h	33 h	68 h

Table 3. Viral integrations identified running ViR with WGS data from *Ae. albopictus* mosquitoes.

nrEVE	Length (bp)	Virus (blastx id %)	Viral protein
nrEVEnew-1	295	KRV (70%)	NS1-NS2A
nrEVEnew-2a	578	KRV (82%)	NS3
nrEVEnew-4/6	534	AeFV (75%)	NS5
nrEVEnew-5	880	KRV (78%)	NS5
nrEVEnew-7	318	AeFV (91%)	E

Supplementary Table 1. Structure of the file “Final_ChimericPairs_Info.txt” which is the output of the script ViR_RefineCandidates. The file will have 18 columns.

Column Name	Column Description
SAMPLE_ID	name of the sample
READ_ID	ID of the pair of reads
HR_CHR	Chromosome in which the host read is mapped
HR_START	Start position of the host read
HR_END	End position of the host read
HR_SEQ	Sequence of the host read
HR_FLAG	Flag of alignment of the host read
HR_NT_BQ20	N. of nt with Phred base quality >20 in the host read
HR_MQ	Phred mapping quality of the host read
VR_SEQ	Sequence of the viral read
VR_FLAG	Flag of alignment of the viral read
VR_NT_BQ20	N. of nt with Phred base quality >20 in the viral read
VIRUS_ID	
VIRUS_START	ID and coordinates of the virus mapping to the viral read
VIRUS_END	
VIRAL_SEQ	viral sequence matching to the viral read
VIRAL_SEQ_LEN	Length of the viral sequence
VR_AlignToRef?	Y/N flag of alignment of the viral read within the host genome

Supplementary Table 2. Viral genome database. Viral species and their sequences are shown.

Viral name,SeqID

Groundnut bud necrosis virus, NC_003620.1,NC_003619.1,NC_003614.1
Murray Valley encephalitis virus, NC_000943.1
Japanese encephalitis virus, NC_001437.1
Venezuelan equine encephalitis virus, NC_001449.1
Dengue virus 2, NC_001474.2
Dengue virus 3, NC_001475.2
Dengue virus 1, NC_001477.1
Onyong-nyong virus, NC_001512.1
Ross River virus, NC_001544.1
Sindbis virus, NC_001547.1
Vesicular stomatitis Indiana virus strain 98COE, NC_001560.1
Tick-borne encephalitis virus, NC_001672.1
Barmah Forest virus, NC_001786.1
Louping ill virus, NC_001809.1
Bunyamwera virus, NC_001927.1,NC_001926.1,NC_001925.1
Yellow fever virus, NC_002031.1
Tomato spotted wilt tospovirus, NC_002052.1,NC_002051.1,NC_002050.1

Dengue virus 4, NC_002640.1
Semliki Forest virus, NC_003215.1
Mayaro virus, NC_003417.1
Sleeping disease virus, NC_003930.1
Peanut bud necrosis virus, NC_003620.1,NC_003619.1,NC_003614.1
Impatiens necrotic spot virus, NC_003625.1,NC_003624.1,NC_003616.1
Modoc virus, NC_003635.1
Rio Bravo virus, NC_003675.1
Apoi virus, NC_003676.1
Powassan virus, NC_003687.1
Langat virus, NC_003690.1
Eyach virus, NC_003707.1,NC_003706.1,NC_003705.1,NC_003704.1,
NC_003703.1,NC_003702.1,NC_003701.1,NC_003700.1,NC_003699.1,NC_003698.1,NC_003697.1,NC_003696.1
Watermelon silver mottle tospovirus, NC_003843.1,NC_003841.1,NC_003832.1
Eastern equine encephalitis virus, NC_003899.1
Aura virus, NC_003900.1
Western equine encephalitis virus, NC_003908.1
Salmon pancreas disease virus, NC_003930.1,NC_003433.1
Tamana bat virus, NC_003996.1
La Crosse virus, NC_004110.1,NC_004109.1,NC_004108.1
Montana myotis leukoencephalitis virus, NC_004119.1
Chikungunya virus, NC_004162.2
Colorado tick fever virus, NC_004191.1,NC_004190.1,NC_004189.1,
NC_004188.1,NC_004187.1,NC_004186.1,NC_004185.1,NC_004184.1,NC_004183.1,NC_004182.1,NC_004181.1,NC_004180.1
Yokose virus, NC_005039.1
Omsk hemorrhagic fever virus, NC_005062.1
Uukuniemi virus, NC_005221.1,NC_005220.1,NC_005214.1
Oropouche virus, NC_005777.1,NC_005776.1,NC_005775.1
Palyam virus, NC_005995.1,NC_005994.1,NC_005993.1,NC_005992.1,
NC_005991.1,NC_005990.1,NC_005989.1,NC_005988.1,NC_005987.1,NC_005986.1
St Croix River virus, NC_006006.1,NC_006005.1,NC_006004.1,
NC_006003.1,NC_006002.1,NC_006001.1,NC_006000.1,NC_005999.1,NC_005998.1,NC_005997.1
African horse sickness virus, NC_006009.1,NC_006019.1,
NC_006018.1,NC_006017.1,NC_006016.1,NC_005996.1,
NC_006012.1,NC_006021.1,NC_006020.1,NC_006011.1
Bluetongue virus, NC_006013.2,NC_006024.2,NC_006025.1,
NC_006023.1,NC_006022.1,NC_006015.1,NC_006014.1,
NC_006010.1,NC_006008.1,NC_006007.1
Sandfly fever Naples virus, NC_006319.1,NC_006320.1,
NC_006318.1,HM566172.1,HM566171.1,HM566170.1,HM566169.1,HM566168.1,HM566167.1,EF201831.1,EF201830.1,EF201828.1
Thogoto thogotovirus, NC_006508.1,NC_006507.1,
NC_006506.1,NC_006504.1,NC_006496.1,NC_006495.1
Usutu virus, NC_006551.1
Getah virus, NC_006558.1
Saint Louis encephalitis virus, NC_007580.2
Yunnan orbivirus, NC_007665.1,NC_007664.1,NC_007663.1,
NC_007662.1,NC_007661.1,NC_007660.1,NC_007659.1,NC_007658.1,NC_007657.1,NC_007656.1
Peruvian horse sickness virus, NC_007757.1,NC_007756.1,
NC_007755.1,NC_007754.1,NC_007753.1,NC_007752.1,NC_007751.1,NC_007750.1,NC_007749.1,NC_007748.1

Capsicum chlorosis virus, NC_008303.1,NC_008302.1,NC_008301.1
Melon yellow spot virus, NC_008307.1,NC_008306.1,NC_008300.1
Entebbe bat virus, NC_008718.1
Sepik virus, NC_008719.1
Aroa virus, NC_009026.2
Akabane virus, NC_009895.1,NC_009894.1,NC_009896.1
West Nile virus lineage 2, NC_009942.1,NC_001563.2
Tomato zonate spot virus, NC_010491.1,NC_010490.1,NC_010489.1
Zika virus strain Natal RGN, NC_012532.1,NC_035889.1
Kedougou virus, NC_012533.1
Bagaza virus, NC_012534.1
Highlands J virus, NC_012561.1
Wesselsbron virus, NC_012735.1
Epizootic hemorrhagic disease virus (serotype 1 / strain New Jersey), NC_013405.1,NC_013404.1,
NC_013403.1,NC_013402.1,NC_013401.1,NC_013400.1,NC_013399.1,NC_013398.1,NC_013397.1,NC_013396.1
Fort Morgan virus, NC_013528.1
Rift Valley fever virus, NC_014395.1,NC_014396.1,NC_014397.1
Great Island virus, NC_014524.1,NC_014529.1,NC_014530.1,
NC_014531.1,NC_014528.1,NC_014527.1,NC_014526.1,NC_014525.1,NC_014522.1,NC_014523.1
Chandiru virus, NC_015374.1,NC_015373.1,NC_015375.1
Sandfly fever Turkey virus, NC_015413.1,NC_015412.1,NC_015411.1
Aguate virus, NC_015452.1,NC_015451.1,NC_015450.1
Groundnut ringspot and Tomato chlorotic spot virus reassortant, NC_015469.1,NC_015468.1,NC_015467.1
Tembusu virus, NC_015843.2
Ndumu virus, NC_016959.1
Southern elephant seal virus, NC_016960.1
Whataroa virus, NC_016961.1
Bebaru virus, NC_016962.1
Bean necrotic mosaic virus, NC_018072.1,NC_018071.1,NC_018070.1
SFTS virus HB29, NC_018138.1,NC_018137.1,NC_018136.1
Aino virus, NC_018465.1,NC_018460.1,NC_018459.1
Sathuperi orthobunyavirus, NC_018466.1,NC_018462.1,NC_018461.1
Shamonda orthobunyavirus, NC_018467.1,NC_018464.1,NC_018463.1
Simbu orthobunyavirus, NC_018478.1,NC_018477.1,NC_018476.1
Ntaya virus, NC_018705.3
Chandipura virus, NC_020805.1
Isfahan virus, NC_020806.1
Wallal virus, NC_022562.1,NC_022561.1,NC_022560.1,NC_022559.1,
NC_022558.1,NC_022557.1,NC_022556.1,NC_022555.1,NC_022554.1,NC_022553.1
Mobuck virus, NC_022629.1,NC_022628.1,NC_022627.1,
NC_022626.1,NC_022625.1,NC_022624.1,NC_022623.1,NC_022622.1,NC_022621.1,NC_022620.1
Razdan virus, NC_022632.1,NC_022631.1,NC_022630.1
Changuinola virus, NC_022642.1,NC_022641.1,NC_022640.1,
NC_022639.1,NC_022638.1,NC_022637.1,NC_022634.1,NC_022636.1,NC_022635.1,NC_022633.1
American bat vesiculovirus TFFN-2013, NC_022755.1
Tyuleny virus, NC_023424.1
Kama virus, NC_023439.1

Arumowot virus, NC_023635.1,NC_023634.1,NC_023633.1
Madariaga virus, NC_023812.1
Cat Que virus, NC_024076.1,NC_024075.1,NC_024074.1
Vesicular stomatitis New Jersey virus, NC_024473.1
Heartland virus, NC_024496.1,NC_024495.1,NC_024494.1
Umatilla virus, NC_024510.1,NC_024509.1,NC_024508.1,NC_024507.1,
NC_024506.1,NC_024505.1,NC_024504.1,NC_024503.1,NC_024500.1,NC_024499.1
Middelburg virus, NC_024887.1
Maraba virus, NC_025255.1
Vesicular stomatitis Alagoas virus, NC_025353.1
Malpais Spring vesiculovirus, NC_025364.1
Yug Bogdanovac vesiculovirus, NC_025378.1
Perinet vesiculovirus, NC_025394.1
Jutiapa virus, NC_026620.1
Cacipacore virus, NC_026623.1
Sokoluk virus, NC_026624.1
Bhanja virus, NC_027142.1,NC_027141.1,NC_027140.1
Wad Medani virus, NC_027547.1,NC_027546.1,NC_027545.1,
NC_027544.1,NC_027543.1,NC_027542.1,NC_027541.1,NC_027540.1,NC_027539.1,NC_027533.1
Chenuda virus, NC_027552.1,NC_027551.1,NC_027550.1,
NC_027549.1,NC_027548.1,NC_027538.1,NC_027537.1,NC_027536.1,NC_027535.1,NC_027534.1
Chobar Gorge virus, NC_027562.1,NC_027561.1,NC_027560.1,
NC_027559.1,NC_027558.1,NC_027557.1,NC_027556.1,NC_027555.1,NC_027554.1,NC_027553.1
Spanish goat encephalitis virus, NC_027709.1
Chrysanthemum stem necrosis virus, NC_027720.1,NC_027719.1,NC_027718.1
Paraiso Escondido virus, NC_027999.1
Cocal virus, NC_028255.1
Potiskum virus, NC_029054.2
Saboya virus, NC_029054.2,NC_033697.1
Spondweni virus, NC_029055.1
Adana virus, NC_029129.1,NC_029128.1,NC_029127.1
Edge Hill virus, NC_030289.1
Tospovirus kiwifruit/YXW/2014, NC_030380.1,NC_030379.1,NC_030378.1
Huangpi Tick Virus 2, NC_031140.1,NC_031139.1,NC_031138.1
Zucchini lethal chlorosis virus, NC_031762.1,NC_031759.1,NC_031758.1
New Mapoon virus, NC_032088.1
Kokobera virus, NC_032088.1,NC_009029.2
Bouboui virus, NC_033693.1
Uganda S virus, NC_033698.1
Jugra virus, NC_033699.1
Meaban virus, NC_033721.1
Gadgets Gully virus, NC_033723.1
Kadam virus, NC_033724.1
Saumarez Reef virus, NC_033726.1
Pepper chlorotic spot virus, NC_033774.1,NC_033773.1,NC_033772.1
Orbivirus SX-2017a, NC_033791.1,NC_033790.1,NC_033789.1,NC_033788.1,NC_033787.1,
NC_033786.1,NC_033785.1,NC_033784.1,NC_033783.1,NC_033782.1

Melon severe mosaic tospovirus, NC_033834.1,NC_033833.1,NC_033832.1
Urucuri virus, NC_033843.1,NC_033842.1,NC_033841.1
Ambe virus, NC_033845.1,NC_033844.1,NC_033835.1
Anhanga virus, NC_033846.1,NC_033837.1,NC_033836.1
Munguba virus, NC_033847.1,NC_033831.1,NC_033830.1
Tapara virus, NC_033848.1,NC_033839.1,NC_033838.1
Uriurana virus, NC_033850.1,NC_033849.1,NC_033840.1
Phnom Penh bat virus, NC_034007.1
Yaounde virus, NC_034018.1
Dhori thogotovirus, NC_034263.1,NC_034262.1,NC_034261.1,NC_034256.1,NC_034255.1,NC_034254.1
Oriboca virus, NC_034488.1,NC_034476.1,NC_034475.1
Caraparu virus, NC_034489.1,NC_034478.1,NC_034477.1
Bwamba orthobunyavirus, NC_034490.1,NC_034480.1,NC_034479.1
Nyando virus, NC_034492.1,NC_034491.1,NC_034481.1
Capim virus, NC_034493.1,NC_034483.1,NC_034482.1
Kaeng Khoi virus, NC_034501.1,NC_034500.1,NC_034499.1
Marituba virus, NC_034504.1,NC_034496.1,NC_034495.1
Madrid virus, NC_034505.1,NC_034498.1,NC_034497.1
Guaroa virus, NC_034506.1,NC_034487.1,NC_034486.1
Morreton vesiculovirus, NC_034508.1
Wolkberg virus, NC_034633.1,NC_034632.1,NC_034631.1
Tomato chlorotic spot virus, NC_035484.1,NC_035483.1,NC_035482.1
Kabuto mountain virus, NC_036606.1,NC_036605.1,NC_036604.1
Calla lily chlorotic spot virus, NC_036609.1,NC_036608.1,NC_036607.1
Zerdali virus, NC_037613.1,NC_037612.1,NC_029901.1
Toros virus, NC_037615.1,NC_037614.1,NC_029903.1
Arrabida virus, NC_037617.1,NC_037616.1,NC_029082.1
Iris yellow spot virus, NC_038232.1,NC_029800.1,NC_029799.1
Carajas virus, NC_038285.1
Piry virus, NC_038286.1
Radi vesiculovirus, NC_038287.1
Watermelon bud necrosis virus, NC_038290.1,NC_038289.1,NC_038288.1
Corriparta virus, NC_038573.1,NC_038572.1,NC_038571.1,NC_038570.1,NC_038569.1,
NC_038568.1,NC_038567.1,NC_038566.1,NC_038565.1,NC_038564.1
Equine encephalosis virus, NC_038583.1,NC_038582.1,NC_038581.1,NC_038580.1,
NC_038579.1,NC_038578.1,NC_038577.1,NC_038576.1,NC_038575.1,NC_038574.1
Eubenangee virus, NC_038593.1,NC_038592.1,NC_038591.1,NC_038590.1,
NC_038589.1,NC_038588.1,NC_038587.1,NC_038586.1,NC_038585.1,NC_038584.1
Lebombo virus, NC_038603.1,NC_038602.1,NC_038601.1,NC_038600.1,
NC_038599.1,NC_038598.1,NC_038597.1,NC_038596.1,NC_038595.1,NC_038594.1
Orungo virus, NC_038613.1,NC_038612.1,NC_038611.1,NC_038610.1,
NC_038609.1,NC_038608.1,NC_038607.1,NC_038606.1,NC_038605.1,NC_038604.1
Warrego virus, NC_038623.1,NC_038622.1,NC_038621.1,NC_038620.1,
NC_038619.1,NC_038618.1,NC_038617.1,NC_038616.1,NC_038615.1,NC_038614.1
Cabassou virus, NC_038670.1
Everglades virus, NC_038671.1
Mucambo virus, NC_038672.1

Pixuna virus, NC_038673.1
Rio Negro virus, NC_038674.1
Tonate virus, NC_038675.1
Alajuela virus, NC_038717.1,NC_038716.1,NC_038715.1
Lukuni virus, NC_038720.1,NC_038719.1,NC_038718.1
Anopheles B virus, NC_038721.1
Batama virus, NC_038722.1
Bimiti virus, NC_038725.1,NC_038724.1,NC_038723.1
Catu virus, NC_038728.1,NC_038727.1,NC_038726.1
Gamboa virus, NC_038731.1,NC_038730.1,NC_038729.1
Guajara orthobunyavirus, NC_038734.1,NC_038733.1,NC_038732.1
Guama virus, NC_038737.1,NC_038736.1,NC_038735.1
Kairi virus, NC_038740.1,NC_038739.1,NC_038738.1
Koongol virus, NC_038743.1,NC_038742.1,NC_038741.1
Mosso das Pedras virus, NC_038857.1
Main Drain virus, NC_038943.1,NC_038942.1,NC_038941.1
Tete orthobunyavirus, NC_039185.1,NC_039184.1,NC_039183.1
Wyeomyia orthobunyavirus, NC_039188.1,NC_039187.1,NC_039186.1
Punta Toro virus, NC_039193.1,NC_039192.1,NC_039191.1
Polygonum ringspot tospovirus, NC_039205.1,NC_039204.1,NC_039203.1
Jurona vesiculovirus, NC_039206.1
Kysanur Forest disease virus, NC_039218.1
Royal Farm virus, NC_039219.1
Kaisodi virus, NC_040494.1,NC_040493.1,NC_040492.1
Oz virus, NC_040735.1,NC_040734.1,NC_040733.1,NC_040732.1,NC_040731.1,NC_040730.1
Alstroemeria yellow spot virus, NC_040743.1,NC_040742.1,NC_040741.1
Rocio virus, NC_040776.1
Ilheus virus, NC_040776.1,NC_009028.2
Guangxi orbivirus, NC_040847.1,NC_040482.1,NC_040481.1,NC_040480.1,
NC_040479.1,NC_040478.1,NC_040477.1,NC_040476.1,NC_040475.1,NC_040474.1,
Thimiri orthobunyavirus, NC_043034.1
Umbre virus, NC_043037.1,NC_043036.1,NC_043035.1
Banzi virus, NC_043110.1
Bukalasa bat virus, NC_043111.1
Carey Island virus, NC_043112.1
Cowbone Ridge virus, NC_043113.1
Dakar bat virus, NC_043114.1
Israel turkey meningoencephalomyelitis virus, NC_043115.1
Koutango virus, NC_043116.1
Sal Vieja virus, NC_043117.1
San Perlita virus, NC_043118.1
Wongorr virus, NC_043192.1
Frijoles virus VP-161A, NC_043209.1,NC_043208.1,NC_043207.1
Groundnut yellow spot virus, NC_043210.1
Trocara virus, NC_043402.1

Una virus, NC_043403.1
M'Poko virus, NC_043431.1
Groundnut ringspot virus, NC_043503.1,NC_043504.1,NC_043502.1
Mukawa virus, NC_043511.1,NC_043510.1,NC_043509.1
Tensaw virus, NC_043548.1,NC_043547.1,NC_043546.1
Peaton virus, NC_043551.1,NC_043550.1,NC_043549.1
Sabo virus, NC_043554.1,NC_043553.1,NC_043552.1
Sango virus, NC_043557.1,NC_043556.1,NC_043555.1
Jamestown Canyon virus, NC_043560.1,NC_043559.1,NC_043558.1
Leanyer virus, NC_043563.1,NC_043562.1,NC_043561.1
Anhembi virus, NC_043566.1,NC_043565.1,NC_043564.1
Iaco virus, NC_043569.1,NC_043568.1,NC_043567.1
Sororoca virus, NC_043572.1,NC_043571.1,NC_043570.1
Cachoeira Porteira virus, NC_043575.1,NC_043574.1,NC_043573.1
Jatobal virus, NC_043578.1,NC_043577.1,NC_043576.1
Batai virus, NC_043581.1,NC_043580.1,NC_043579.1
Schmallenberg virus, NC_043584.1,NC_043583.1,NC_043582.1
Ilesha virus, NC_043587.1,NC_043586.1,NC_043585.1
Ingwavuma virus, NC_043590.1,NC_043589.1,NC_043588.1
Mermet virus, NC_043593.1,NC_043592.1,NC_043591.1
Utinga virus, NC_043596.1,NC_043595.1,NC_043594.1
Buttonwillow virus, NC_043599.1,NC_043598.1,NC_043597.1
Pacui virus, NC_043602.1,NC_043601.1,NC_043600.1
Rio Preto da Eva virus, NC_043605.1,NC_043604.1,NC_043603.1
Guertu virus, NC_043611.1,NC_043610.1,NC_043609.1
Enseada virus, NC_043614.1,NC_043613.1,NC_043612.1
Fort Sherman virus, NC_043617.1,NC_043616.1,NC_043615.1
Cache Valley virus, NC_043620.1,NC_043619.1,NC_043618.1
Keystone virus, NC_043629.1,NC_043628.1,NC_043627.1
Lumbo virus, NC_043632.1,NC_043631.1,NC_043630.1
Melao virus, NC_043635.1,NC_043634.1,NC_043633.1
San Angelo virus, NC_043638.1,NC_043637.1,NC_043636.1
Serra do Navio virus, NC_043641.1,NC_043640.1,NC_043639.1
Potosi virus, NC_043647.1,NC_043646.1,NC_043645.1
Birao virus, NC_043652.1,NC_043651.1,NC_043650.1
Bozo virus, NC_043655.1,NC_043654.1,NC_043653.1
Laurel Lake virus, NC_043681.1,NC_043680.1,NC_043679.1
Zegla virus, NC_043688.1,NC_043687.1,NC_043686.1
Patois virus, NC_043691.1,NC_043690.1,NC_043689.1
Tacaiuma orthobunyavirus, NC_043694.1,NC_043693.1,NC_043692.1
Shuni orthobunyavirus, NC_043699.1,NC_043698.1,NC_043697.1
Koyama Hill virus, AB894493.1,AB894492.1,AB894491.1,AB894490.1,AB894489.1,
AB894488.1,AB894487.1,AB894486.1,AB894485.1,AB894484.1,
Cholul virus, JN808310.1,EU879062.3,
Salt ash virus, KF234258.1,KF234257.1

El Huayo virus, KT334540.1
Anadyr virus, KU159766.1,KU159765.1,KU159764.1
Parry's Lagoon virus, KU724119.1,KU724118.1,KU724117.1,KU724116.1,KU724115.1,
KU724114.1,KU724113.1,KU724112.1,KU724111.1,KU724110.1
Calbertado virus, KX669682.1
Long Pine Key virus, KY290249.1
La Tina virus, KY320649.1
Marisma mosquito virus, MF139576.1
CFLV NEI 1, M91671.1
Anopheles hinesorum orbivirus, MF073202.1
Anopheles annulipes orbivirus, MF281534.1
Skunk River virus , MK100578.1,MK100577.1,MK100576.1,MK100575.1,MK100574.1,
MK100573.1,MK100572.1,MK100571.1,MK100570.1,MK100569.1
Cell fusing agent virus, NC_001564.2
Kamiti River virus, NC_005064.1
Culex flavivirus, NC_008604.2
Quang Binh virus, NC_012671.1,NC_021069.1
Stretch Lagoon orbivirus, NC_012755.1,NC_012754.1
Aedes flavivirus, NC_012932.1
Donggang virus, NC_016997.1
Chaoyang virus, NC_017086.1
Eilat virus, NC_018615.1
Mosquito flavivirus, NC_021069.1
Brazoran virus, NC_022039.1,NC_022038.1,NC_022037.1
Murrumbidgee virus, NC_022597.1,NC_022596.1,NC_022595.1
Fengkai orbivirus, NC_027811.2,NC_027815.1,NC_027814.1,NC_027813.1,
NC_027812.1,NC_027807.1,NC_027806.1,NC_027805.1,NC_027804.1,NC_027803.1
Tibet orbivirus, NC_027811.2,NC_027815.1,NC_027814.1,NC_027813.1,
NC_027812.1,NC_027807.1,NC_027806.1,NC_027805.1,NC_027804.1,NC_027803.1,
Parramatta River virus, NC_027817.1
Mercadeo virus, NC_027819.1
Hanko virus, NC_030401.1
Tai Forest alphavirus, NC_032681.1
Palm Creek virus, NC_033694.1
Nounane virus, NC_033715.1
T'Ho virus, NC_034151.1
Ochlerotatus caspius flavivirus, NC_034242.1
Nanay virus, NC_040610.1
Culex theileri flavivirus, NC_040682.1
Kampung Karu virus, NC_040788.1
Culex pipiens pallens densovirus, EF579756.1
Culex densovirus 0507JS11, FJ805445.1
Culex Y virus, JQ659255.1,JQ659254.1,
Dezidougou virus, JQ675604.1
Santana virus, JQ675606.1
Mosquito X virus, JX403942.1,JX403941.1,

Kamphang Phet virus, KC807172.1
Culicine-associated Z virus, KF298272.1,KF298271.1
North Creek virus, KF360973.1,KF360972.1
Mosinovirus, KJ632943.1,KJ632942.1
Wuhan Mosquito Virus 3, KM817622.1
Wuhan Mosquito Virus 5, KM817624.1
Wuhan Mosquito Virus 7, KM817626.1
Jiangxia Mosquito Virus 1, KM817670.1
Jiangxia Mosquito Virus 2, KM817671.1
Xinzhou Mosquito Virus, KM817701.1
Qingnian Mosquito Virus, KM817710.1,KM817673.1
Terena virus, KT966493.1
Anopheles Cypovirus, KU169880.1,KU169879.1
Anopheles flavivirus variant 2, KX148547.1
Anopheles totivirus, KX148550.1
Bolahun virus variant 1, KX148551.1
Bolahun virus variant 2, KX148552.1
Gambie virus, KX148553.1
Chaq virus-like1, KX148554.1
Chaq virus-like2, KX148555.1
Chaq virus-like3, KX148556.1
dsRNA virus-like 1, KX148558.1
dsRNA virus-like 2, KX148559.1
dsRNA virus-like 3, KX148560.1
dsRNA virus-like 4, KX148561.1
dsRNA virus-like 5, KX148562.1
dsRNA virus-like 6, KX148563.1
dsRNA virus-like 7, KX148564.1
dsRNA virus-like 8, KX148565.1
dsRNA virus-like 9, KX148566.1
Endornavirus-like 1, KX148567.1
Endornavirus-like 2, KX148568.1
Endornavirus-like 3, KX148569.1
Mononegavirus-like 1, KX148570.1
Mononegavirus-like 2, KX148571.1
Mononegavirus-like 3, KX148572.1
Mononegavirus-like 4, KX148573.1
Mononegavirus-like 5, KX148574.1
Partitivirus-like 3, KX148577.1
ssRNA virus-like 1, KX148580.1
ssRNA virus-like 2, KX148581.1
ssRNA virus-like 3, KX148582.1
ssRNA virus-like 4, KX148583.1
ssRNA virus-like 5, KX148584.1

ssRNA virus-like 6, KX148585.1
Guaico Culex virus, KT966499.2, KM461670.1, KM461669.1, KM461668.1, KM461667.1, KM461666.1
Hubei sobemo-like virus 39, KX882828.1
Hubei noda-like virus 12, KX883125.1
Hubei partiti-like virus 22, KX883195.1
Wuhan Mosquito Virus 4, KX883867.1, KX883866.1, KX883865.1, KM817623.1
Shuangao partiti-like virus 1, KX884100.1
Sinu virus, KX949591.1, KX949590.1, KX949589.1, KX949588.1, KX949587.1, KX949586.1
Panmunjeom flavivirus, KY072986.1
Omono River virus, KY264024.1
Ohlsdorf virus, KY768856.1
Bontang Baru virus, MF158348.1
Culex phasma-like virus, MF176244.1, MF176243.1, MF176242.1
Ngewotan virus, MF176279.1
Aedes alboannulatus reo-like virus, MF176328.1, MF176327.1, MF176326.1, MF176325.1
Shuangao chryso-like virus 1, MF176343.1, MF176342.1, MF176341.1, MF176340.1,
Wilkie qin-like virus, MF176289.1, MF176290.1, MF176349.1, MF176350.1
Culex mononega-like virus 1, MF176316.1, MF176317.1
Wuhan Mosquito Virus 6, MF176383.1, MF176382.1, MF176381.1, MF176380.1, MF176379.1, MF176378.1
Culex Luteo-like virus, MF176386.1, MF176387.1
Hubei chryso-like virus 1, MF176391.1, MF176390.1, MF176389.1, MF176388.1
Lobeira virus, MF344595.1, MF344594.1, MF344593.1
Merida-like Turkey virus, MF882997.1
Barkedji virus, MG214905.1
Houston virus, MH719099.1
Sabethes flavivirus, MH899446.1
Hubei reo-like virus 7, MK133924.1
Armigeres subalbatus virus SaX06-AK20, NC_014609.1
Culex originated Tymoviridae-like virus, NC_018703.1
Anopheline-associated C virus, NC_023682.1, NC_023683.1,
Anopheles minimus iridovirus, NC_023848.1
Tanay virus, NC_024071.1,
Mosquito Circovirus, NC_026512.1
Anopheles flavivirus variant 1, NC_031327.1
Zhejiang mosquito virus 3, NC_032491.1
Big Cypress virus, NC_034152.1,
Biratnagar virus, NC_034153.1,
San Bernardo virus, NC_034154.1,
Cordoba virus, NC_034156.1,
Fort Crockett virus, NC_034157.1,
Wilkie Partiti-like virus 2, NC_035119.1
Wilkie narna-like virus 2, NC_035120.1
Point Douro narna-like virus, NC_035121.1
Wilkie Partiti-like virus 1, NC_035122.1

Culex Negev-like virus 2, NC_035123.1
Culex Negev-like virus 1, NC_035124.1
Leschenault Partiti-like virus, NC_035125.1
Wilkie narna-like virus 1, NC_035126.1
Aedes camptorhynchus negev-like virus, NC_035128.1
Culex Negev-like virus 3, NC_035129.1
Aedes alboannulatus toti-like virus, NC_035130.1
Culex rhabdo-like 1, NC_035132.1
Culex mononega-like virus 2, NC_035133.1
Aedes camptorhynchus reo-like virus, NC_035223.1, NC_035222.1, NC_035217.1, NC_035216.1
Bellavista virus, NC_043623.1, NC_043622.1, NC_043621.1
African swine fever virus, NC_044944.1

Fig. 1

Fig. 2

Fig. 3

Sup. Fig. 1