**Separation and assembly of deep sequencing data into discrete sub-population genomes**

Konstantinos Karagiannis1, Vahan Simonyan2, Konstantin Chumakov2 and Raja Mazumder1,3\*

1Department of Biochemistry and Molecular Medicine, George Washington University Medical Center, Washington, DC 20037.

2Center for Biologics Evaluation and Research, Food and Drug Administration, Silver Spring, MD 20993, USA.

3McCormick Genomic and Proteomic Center, George Washington University, Washington, DC 20037, USA

\* Corresponding authors.

Mailing address:

Department of Biochemistry and Molecular Medicine

The George Washington University

Ross Hall, Room 540

2300 Eye Street N.W.

Washington, DC 20037

Phone: 202-994-5004

E-mail: mazumder@gwu.edu

**Abstract**

Sequence heterogeneity is a common characteristic of RNA viruses that is often referred to as sub-populations or quasispecies. Traditional techniques used for assembly of short sequence reads produced by deep sequencing, such as *de-novo* assemblers, ignore the underlying diversity sub-populations genomes. Here, we introduce a novel algorithm that simultaneously assembles discrete sequences of multiple genomes present in populations. Using *in silico* data we were able to detect populations at as low as 0.1% frequency with complete global genome reconstruction and in a single sample detected 16 resolved sequences with no mismatches. We also applied the algorithm to high throughput sequencing data obtained for viruses present in sewage samples and successfully detected multiple sub-populations and recombination events in these diverse mixtures. High sensitivity of the algorithm also enables genomic analysis of heterogeneous pathogen genomes from patient samples and accurate detection of intra-host diversity, enabling not just basic research in personalized medicine but also accurate diagnostics and monitoring drug therapies, which are critical in clinical and regulatory decision-making process.

**Background**

Genetic diversity of populations resulting from high mutation rates plays a key role in viral evolution. In extreme cases, mutation rate can be as high as 4×10-4 errors[1] per nucleotide per round of replication (RNA viruses such as HIV, poliovirus, etc.). Therefore, natural virus populations consist of an extremely high number of micro-variants, often referred to as sub-populations or quasispecies[2]. It is well known that such genetic diversity resulting from both point mutations and recombination events is critical for maintaining fitness of the virus[3]. Evolution of viruses is driven by selection from the preexisting universe of these variants, in response to changing replication conditions and various pressures such as the immune system, drug treatment, switching to another host, etc. Therefore, identification of population heterogeneity is of critical importance for treatment design[4, 5] and pathogen surveillance[6] .

Traditional approaches, such as Sanger sequencing[7], are not capable of addressing the sub-population issue because they are geared towards sequencing DNA from homogenous and pure samples. High-throughput (deep) sequencing technologies (HTS)[8, 9] that produce highly redundant (massively parallel) sequencing information are already used in clinical diagnostics[10] and can provide the necessary level of detail for sub-population genome delineation. However, specialized analysis is required to interpret properly the genetic diversity. In recent years, few algorithms have been designed to use HTS information, but accurate population reconstruction and frequency estimation is still somewhat intangible[11-15].

Available tools address the problem using statistical approaches, applying error correction filters to the reads, developing *de novo* assembles for diverse samples[16], building graphs from overlapping reads, or utilizing expectation maximization algorithms to reconstruct either local or global sequences[17-22]. The accuracy of reconstruction is affected by the heterogeneity of intra-host viral population. Abundance of conserved genomic regions that extend significantly beyond the maximal read length restricts the full-genome assembly of highly heterogeneous populations[11]. Nevertheless, none of the current algorithms can efficiently process current HTS data produced in deep-sequencing experiments. More specifically, ShoRAH[22] performs multiple sequencing alignment and clustering of limited number of reads (up to tens of thousands) and calls haplotypes based on the centers of the clusters. ViSpA[21] and QuRe[18] were designed for reads generated by pyrosequencing technologies with techniques to address insertions, deletions and errors in homopolymeric regions and therefore cannot adequately handle current HTS data. Finally, PredictHaplo[19] and HaploClique[17] have been tested with technologies other than pyrosequencing, including MiSeq paired end reads but still like ShoRAH have limitations on the number of reads it can analyze.

Here, we describe a novel deterministic algorithm, called Hexahedron, based on HTS data that can reconstruct local and global sequences and determine their relative frequency at a much larger scale than what was manageable before. We also offer a novel visualization technique that comprehensively represents the dynamic nature of the results with a simple interactive interface. The graph contains, per position depth of coverage, information for each reconstructed sequence and the first (bifurcation) and last (merging) position that differentiates a sequence and its closest sequence from which it has been derived. Annotations of the provided references, if included in the input set, will also be plotted in the same coordinate system as the reconstructed sequences. Finally, the interface allows sequences to be combined with each other following the paths that connect different bifurcation and merging positions as well as their consensuses and path compositions to be extracted.

**Results**

Hexahedron is a deterministic algorithm that extensively explores all mutations above a given threshold in all available genomes in a sample. Each position is treated individually so that multiple nucleotide variations are processed as a series of single nucleotide variations (SNVs). An SNV is defined within the context of comparing a particular sequence read with the consensus sequence of multiple references; thus, sub-populations that have consistent groups of mutated nucleotides will be detected as correlated groups of SNV patterns in that specific sub-population or quasispecies genome. The input of the algorithm is the result of the alignment of the sequencing reads against a group of related references. Although the algorithm can accept a single reference, it is beneficial to use a multiple sequence alignment of related genomes to avoid the bias of reference selection. A comprehensive set of multiple references covering the entire range of sequence variations provides a better scaffold for reference assisted *de novo* assembly of the heterogeneous genomes in the sample. Allowing multiple references also provides the advantage of aligning more reads that would, otherwise, be too distant to match a single sequence and therefore be omitted from analysis (Table 1, Fig S1-9). Each read is aligned with the highest scoring reference sequence and then the result is re-mapped to a common frame generated by a multiple alignment of the reference sequences to each other. This provides the best scoring alignment to the frame of the common coordinate system. The algorithm proceeds in a step-wise manner following a 5’ to 3’ directionality of the references. In every step the variant calling profile along the reference frame is constructed and mutations trigger branching of the variant calling profile and re-allocation of the alignments. Besides requiring a collection of alignments, there are two additional user defined parameters: the frequency of occurrence threshold, above which a mutation is considered valid, and optionally the mutual alignment of reference sequences in case of alignment against multiple references (Fig. 1).

**Algorithm description**

The algorithm starts by constructing a frame (window) with a dynamic size. The frame is placed in the “leftmost” position of the reference (5’-end) and its size is set at twice the length of the first alignment. Then, all the alignments starting within this frame are scanned and a SNV profile is constructed. The profile extends beyond the original frame and occupies twice the size of the longest alignment considered so far. At the end of this step, the frame is scanned for mutations above the given threshold and a new quasispecies genome profile is bifurcated based on the first mutation. The alignments are then re-assigned to the appropriate branches based on the mutation that triggered this bifurcation event. This process continues recursively until all frames are clean of mutations exceeding the threshold. In the next step, the extended profile of the frame serves as an overlap region with the next step, which is used to determine to which frame each alignment belongs. During the reconstruction, the information about the alignments and contribution of each reference in any given position for any given frame is maintained. This leads to resolved profiles, where each one also represents a similarity plot based on the reference selected during the alignment step. The complexity of the algorithm is where is the number of short reads and the number of mutations that trigger bifurcations.

**Validation using *in silico* reads**

The algorithm’s performance was validated using simulated *in-silico* data and determined the sensitivity and specificity boundaries. In addition, we have applied the algorithm to real samples and analyzed the results using the novel interactive visualization, similar to Sankey diagrams but developed specifically for this purpose. Each band represents a separate reconstructed sequence and the width of the flow is represents the depth of the coverage for each position along the x-axis. It comprehensively represents where a contig is detected relative to the common coordinate system, the coverage of that contig and finally where and how the contig ends. Bifurcation and merging events are represented by grey line. Additionally, different colors inside each contig describe the similarity at a given position based on the references to which the reads, considered for the position, have been aligned. In the Sankey diagram, all trajectories following the bifurcation and merging events are possible assemblies in the variant spectrum.

First, the validity of the concept was tested by generating a mixture of reads derived from ten sequences (Table S1). Despite the fact that the sequences were detected as two groups of variants of the same frequency, the assembler identified all of them (Fig S10). In all experiments, reads were aligned using the Hexagon aligner[23]. The quality of the results was measured based on precision, recall and Kullback-Leibler[24] divergence.

Sensitivity: To determine the sensitivity of the algorithm, two sequences 20% different from each other were produced, split into short reads of 250bp and combined together in 6 different proportions (Table S2, S3). The reads were aligned against the reference sequences used to generate the two original sequences, and processed by the algorithm. One percent mutation threshold was selected for the first 3 samples, 0.3% for the next 2 and 0.07% for the last. Threshold was set so that it lies above the expected 0.3% noise level, unless we specifically wanted to detect populations below the noise level. The assembled sequences were then aligned back to the original references and those that were aligned only against their original sequence were reported as true positives (TP).

Filtering short contigs also decreased false positives generated by phased noise. All datasets resulted in fully reconstructed sequences with 100% recall (Table 2 and Fig S11). After filtering all contigs shorter than 2000bp, precision was also 100% for all samples. It remained 100% even when contigs greater than 1000bp were allowed with the exception of the 0.1% dataset, where 24 false positives (FP) were reported having a great impact in the precision. It is understandable that the noise, which in this case is three times higher than the bifurcation threshold, was phased allowing contigs to extend up to 1500bp. No mismatches were detected between the true positives and the original sequences confirming sequence reconstruction accuracy. The distribution of the predicted sequences was identical to the actual distribution with divergence at the level of 10-5 (Table 2).

Specificity: To determine the specificity, the closest genetic distance the algorithm can separate sequences, a number of additional datasets were generated (Table S4) with an increasing number of sequences (2, 4, 8, 16, 32 and 64) derived from Sabin 2 poliovirus[25] (GenBank accession: AY184220.1). Each sample exhibited the same diversity, such that the two marginal sequences of each sample had almost the same distance across datasets, resulting in sequences that were genetically very close to each other (Tables S5-S10). Furthermore, the relative frequency of each sequence was calculated based on the exponential distribution resulting in sub-populations of frequency as low as 0.05% (see Methods).

Analysis of the samples with 2, 4, 8 and 16 sequences (with frequencies as low as 0.25%) led to global reconstruction of all sequences (Table 2). The recall and the precision for these samples was 100% with no FP contigs detected longer than 300bp (Fig S12a-d and Fig S13a-d). All globally reconstructed sequences were aligned against the original sequence and found to have less than 0.5% mismatches (Additional File 5). This occurred in low-coverage regions such as the 5´ and 3´ends. After trimming these regions out of the predicted sequences, the number of mismatched position decreased to less than 0.04% for these samples (data not shown). The distribution of frequencies of the predicted sequences for these samples was identical to the true distribution with KL divergence 10−5. Recall and precision decreased for samples with 32 and 64 sequences (Table 2). Twenty-seven fully resolved sequences were generated for the mixture of 27 sequences and 39 for the mixture of 64 (Fig S12e-f and Fig S13e-f). Prediction of relative frequencies was also affected in these dense and highly heterogeneous samples. This was observed mainly for the low frequency haplotypes, while those with higher frequency resulted in a more accurate prediction (Additional File 5). This may be because a large number of the generated sequences were present below the noise level.

Speed performance: To measure the speed performance of the algorithm (see Additional File 1, Fig S14), we constructed different sets of samples changing all pairs of the following characteristics: the noise, the number of sequences and the distance between sequences. Noise was set to 0.3%, number of sequences to 4, distance between them to 2%, number of short reads to 100,000 and the read length to 200 as defaults and the mutation threshold to 1%. The algorithm was shown to perform well even on deep sequencing data and most of the computations finished within seconds (Fig S14). The slowest computations, finishing within 1hr, were the ones with the greatest number of aligned reads and when sensitivity was set below the noise baseline.

**Validation using experimental data for viral samples with sub-populations**

In addition to the above simulation study, the algorithm was applied to real HTS datasets obtained for a live attenuated vaccine strain and for an environmental (sewage) sample. All samples were aligned using hexagon; when necessary, multiple alignment of selected references was performed using MAFFT[26].

Mumps virus. Mumps is an RNA virus with a 15000bp genome that encodes nine proteins. Jeryl Lynn strain used in live Mumps virus vaccine was shown by conventional sequencing of plaque-purified clones to contain two distinct virus sequences[27]. After aligning the paired end reads to a comprehensive database of genomic sequences of 54 strains of mumps, using hexagon with default parameters, 688,000 hits were recorded and used for analysis. Two globally reconstructed sequences were detected using 2% mutation threshold (Fig. 2a). The predicted frequencies of the reconstructed sequences were 93.18% and 6.82% (Fig.2b), consistent with previous estimate based on quantitative PCR[27]. Consensus sequences of these two sub-strains were identical to those determined by conventional sequencing of plaque-purified clones.

Environmental isolate of poliovirus (example 1): The same analysis was applied to a sewage sample that was previously found by conventional virological analysis to contain vaccine poliovirus along with another non-polio enterovirus. The 55 \*106 paired end reads were aligned against Sabin1, Sabin2, and Sabin3 sequences (Genbank accession numbers AY184219, AY184220, and AY184221 respectively) resulting in 48.26 million alignments. The reads produced hits to all of the references and the alignments were used as input for our algorithm with a 5% mutation threshold. Two sequences were reconstructed, one globally and another of 5948 nucleotides long (Fig. 2c and Additional File 6). The globally reconstructed sequence was also the dominant one and had a predicted frequency of 54.75%; the shorter sequence had a predicted frequency of 45.25% (Fig. 2d and Additional File 6). The shorter sequence bifurcates from the dominant sequence at position 53 and merges back at position 5981 of the common coordinate system. Hence, the sample contains a mixture of two recombinant viruses that differ only in this range by 2513 mutations (Additional File 6). Comparison of the reconstructed consensus sequences showed that one component of the mixture was a Sabin 2 - Sabin 3 recombinant, while another was close to Echovirus 11.

Environmental isolate of poliovirus (example 2): The same analysis was applied to another virus isolated from sewage. The 2 \*106 paired end reads were aligned against a comprehensive set of 500 enteroviruses resulting in 3 million alignments. The reads aligned produced hits to 315 of the references and used as an input for our algorithm with a 1% mutation threshold. The large number of references did not affect the efficiency of the algorithm; it did, however, affect the mutual alignment frame, where we observed an increased number of gaps (Fig. 2e). The algorithm identified two major fully reconstructed variants, and several minor variants that represented recombinants of the first two with different inserts with length between 1000bp and 5000bp. The two fully reconstructed sequences have predicted frequencies of 64% and 26%, and are identified as recombinant vaccine-derived polioviruses of serotypes 1 and 3. (Fig.2d and Additional File 6). In the set of minor variants, frequencies below the 1% mutation threshold were detected (0.98%), demonstrating the flexibility of the algorithm.

**Comparative studies**

Hexahedron was also compared against the current state-of-the-art sequence reconstruction algorithms compatible with the available computational environment (see Methods): QuRe[18], ViSpA[21] ShoRah[28] and PredictHaplo[29, 30]. Initially, 20 samples were generated varying in sequence length of the populations with 500, 1000, 2000 and 5000 base pairs and the number of the *in silico* generated short reads with 1000, 5000, 10000, 50000 and 100000 short reads. All samples were simulated to consist of four populations each one at 25% prevalence. The sequences were derived from Sabin 2 poliovirus (see Methods) with an increasing distance from the original template of 1% positions on average (Table 3).

Reconstruction performance: The reconstructed haplotypes were compared to the original sequences and were called True Positive (TP) or False Positive (FP) depending on whether the percentage of mismatches exceeds one of the following thresholds: 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5%. Hexahedron produced the least amount of FP and False Negative (FN) with few exceptions, only against PredictHaplo (Fig S18, S19). All tools failed to reconstruct any haplotype with less than 0.5% mismatches for samples with average depth of coverage less than 5x per population (Table 3). ShoRah, QuRe and ViSpa failed to reconstruct any haplotype with less than 1% mismatches for any of the samples with the only exception of ViSpa that successfully reconstructed 2 sequences for the sample of the 1000 short reads and 500 bp long sequences. In contrast, for samples with more than 10000 reads and considering a 0.1% mutation threshold Hexahedron produced consistently the least amount of FP and FN and as a result it performed better against all algorithms in terms of precision (Fig S16), recall (Fig S7) and F-score (Fig 3). Using the F-score to compare the algorithms, PredictHaplo performed better in samples SCS7, SCS8 and SCS19 only when considering 0.05% as mismatch threshold, in samples SCS13 and SCS20 with 0.05% and 0.1% mismatches and in SCS12 for thresholds between 0.01% and 1. Furthermore, PredictHaplo did not reconstruct perfectly any sequence, while Hexahedron produced all sequences without mismatches in sample SCS1, SCS2, SCS3, SCS5, SCS9 and SCS10 and perfectly reconstructed at least one sequence in samples SCS4, SCS14 and SCS15.

Phylogeny of reconstructed sequences: In addition to diagnostic testing, neighbor-joining relatedness trees were constructed for each sample including the original and all sequences predicted by all tools (Fig 4 and Fig S20). The tree of sample SCS5 is represented in Figure 4a and is in concordance with the F-scores reported in Figure 3a. Hexahedron predicted exactly four sequences while PredictHaplo reconstructed one more than expected with 1% abundance, possibly due to noise and high depth of coverage (12500 per sequence). Despite similar F-scores in sample SCS11, Figure 4b shows that, PredictHaplo predicted only two sequences, each one with almost 50% abundance, while Hexahedron predicted exactly four with roughly 25% abundance each. Consequently, this difference was also highlighted when we measured the accuracy of the detected abundances of the reconstructed sequences. For this purpose, we measured the Jensen-Shannon divergence between the predicted and the original distribution. Each original sequence was paired with the closest reconstructed one without considering any mismatch threshold. PredictHaplo predicted more accurately the frequency distribution than Hexahedron only in samples SCS16 and SCS17 while the latter outperformed all the tools in samples with 100000 reads with the only exception of sample SCS20, where PredictHaplo achieved the same accuracy. In sample SCS16, Hexahedron reconstructed seven sequences and PredictHaplo reconstructed one instead of four, while in SCS17 Hexahedron reconstructed five sequences and PredictHaplo four resulting in more accurate frequency distribution prediction.

Speed performance: Finally, in terms of speed, Hexahedron was found to be the fastest for all samples (Fig. 3c). In fact, all other tools displayed a polynomial time complexity with respect to the number of short reads with the only exception of samples SCS17 and SCS18, where QuRe spent more time to process the sample with smaller number of reads (SCS17). Notably, QuRe predicted one sequence for SCS17 and two for SCS18. Perhaps the depth of coverage in SCS17 forced QuRe to consider more reconstructed sequences per sliding window, but this was not sufficient to expand them into global sequences providing the worst-case scenario for the algorithm. Conversely, SCS18 with sufficient depth of coverage allowed QuRe to reconstruct two sequences, removing a number of variants from the single pool of the four populations and reducing the computational load.

Large datasets: In order to further compare the performance of the tools on datasets more representative of current HTS technologies, we generated four more datasets with 16 populations with abundance following a power distribution, similar to sample SP4 (Table 3 and Table S12) and with number of reads ranging from 100,000 to 5,000,000. Each tool was given up to 4 days of wall clock time to conclude the computations (Table 3). Within this timeframe, only Hexahedron processed successfully all samples and PredictHaplo successfully finished only samples SCL1, SCL2 and SCL3. None of the sequences reconstructed by PredictHaplo was identified as a TP even considering mismatch threshold as high as 20%. Hence, the results were not analyzed further (Table S13, Fig S16 and Additional File 3). Hexahedron was able to reconstruct successfully all haplotypes with (Additional File 7).

**Discussions**

Deep sequencing is a powerful tool for the analysis of heterogeneous populations that are present in most specimens derived from natural sources (environment, clinical isolates) and those that emerge during treatment of viral diseases and cancer. Creation of new experimental and mathematical techniques, as well as optimization of the existing protocols will play a crucial role in discovering and interpreting the biological impact of the sequence heterogeneity. It could be used for the surveillance for viral pathogens as well as monitoring the emergence of drug-resistant variants during the treatment of infectious diseases and cancer. This study presents a new algorithm that identifies discrete populations in heterogeneous samples based on mutation patterns in the genetic profile. Due to the nature of the algorithm, the only limiting factor is the size of the step, which is defined by the length of the alignments. Thus, the performance of the algorithm improves as the sequencing technologies advance and are able to produce longer reads with greater accuracy. *In silico* experiments with longer reads (Additional File2 S1-S4) demonstrated an increase in the accuracy of the sequence reconstruction; however short reads produced by current technologies appear to be already sufficient to separate sub-populations. Inaccurately reconstructed sequences and mismatched nucleotides were mostly observed in short contigs. One explanation is that the noise was partially phased with the mutation patterns that identified the sequence. This might cause the noise to be included in reconstructed sequences (particularly in the low frequency populations) or trigger the bifurcation of the contigs, which in turn accumulates the defining mutations of the sequence.

Importantly, the number of the selected references does not necessarily affect the sensitivity of the analysis. A large set of homologous references, such as the comprehensive list of enteroviruses used for the environmental samples, versus a small set of references that sufficiently maps all short reads will produce the same results even if the original sequences are absent in the reference set (Table 1). The density, with which the genetic space needs to be covered, is specified by the parameters of the alignment. In fact, comprehensive reference sets allow for identification of more distant and diverse populations without increasing the ambiguity with flexible alignments. Importantly, the information of the aligned reference is maintained and displayed for each reconstructed contig at the end of the process. At this point, the density with which this space has been covered by the selected references is indicated by the number of references contributing at every position. This is not a recombination analysis, where the identified references are part of recombination events. The graph is offering an overview of the closest similarities with references across the reconstructed sequences and serves as an indicator for a further, more detailed, recombination analysis. It is true though that careful selection of references can lead to robust results in terms of recombination events.

The algorithm combined with the Sankey diagram of the reconstructed contigs gives a comprehensive representation of the genotypic cloud that describes highly diverse, viral populations. This study proposes a novel approach that can capture the mutant spectrum of evolving diverse populations in an exceptional accuracy and an unparalleled speed. Both characteristics will eventually be vital in clinical research where antiviral efficacy and recent combinatorial treatments have already proved[31, 32] to be affected by escape mutants. Recent studies on vertically HIV infected children after failed nevirapine prophylaxis (NVP) revealed the existence of linked multiclass drug resistant mutations using single genome sequencing [33]. Hence, our approach is of critical significance and can help to address the challenge of discovering broad quasispecies spectrum that single genome sequencing is unable to do due to its low throughput. From a broad basic research perspective, it will enhance the arsenal of supersensitive genotyping methods allowing the evaluation of superinfection, major variants and minor variants. Finally, we anticipate that this approach could be extremely useful for virological surveillance that is vitally important for timely identification of emerging pathogens and development of rational countermeasures.

**Methods**

**Algorithm**

Let denote the reference sequences we used and the mutual alignment between these sequences. Let be the position of the reference so be the position of the mutual alignment. Similarly, let be the short read aligned to reference sequence and be the reference coordinate that the position of the short read was aligned to. Let also be the short read mapped to . Finally let be the contig, the position of and the basecall for any position.

Initially, we map all reads aligned against all references to the mutual alignment coordinates, so that:

where is a fixed look up table of the specific that returns the coordinate of the mutual alignment that corresponds to .

Then all short reads are sorted based on their aligned position on increasing order so that . The sorting method used is quicksort. Let be the matrix where the contig will be constructed into and the position of . Hence, is the total number of Adenines, Cytosines, Guanines and Thymines; insertions and deletions are mapped in . As a result, during the first step of the algorithm:

Where and. However, the length of is dynamically allocated during this step, and self-adjusted to the maximum length of the short reads that have been scanned so far.

After all that fall under the range of the first step have been scanned and the initial contig has been constructed, we check for mutations that exceed the frequency threshold (). So:

Therefore, we consider the bifurcation position (or in the context of the ) where:

After a bifurcation position has been detected, we construct a new matrix and iterate the contig construction process using a subset of all reads mapped through the position p and containing the bifurcating mutation. Now is in the range where . Another difference is that now we have to decide the matrix the will contribute to. The bifurcation and construction steps are iterated until no bifurcation position is found.

With the exception of the first step of the algorithm, where there is only one contig, a decision is made before assignation of a short read to a contig. There are different points in the algorithm where contig voting is applied. The first is after bifurcation; the decision is between contig where the bifurcation position was detected and contig the newly created one. The decision is made based on the base of the bifurcated position. So:

The second scenario is after the end of the first step of algorithm, where there are an arbitrary number of frames. In this case:

Where and . Where is the Hamming distance between the read and the contig.

Before each step, all contigs are scanned to ensure that they are well supported by mutated positions against all other contigs. As the algorithm proceeds, all contigs unsupported for a region greater than the size of the step are merged together. Similarly, contigs that drop the coverage inside the current frame to zero - are excluded from participation in the construction process. So

Where and is the Hamming distance between the read and the contig for position .

The algorithm iterates until the last alignment is considered on the rightmost frame. The result is a collection of arrays, each representing a contig . Position of the contigis an array where and Pr is the number of alignments that support the contig and have been aligned to reference .

**Validation**

In order to test the algorithm, we generated in silico short reads using Sabin2 as a template sequence (Table S1). One sequence was used to generate ten mutant strains each with a genetic distance described in the second column of Table S1. From these sequences, we randomly generated different number of short reads creating different relative frequencies for each haplotype. The length of the reads and the random noise introduced in each group of short reads is also described in the table. All short reads were merged into one file. After creating the in silico heterogeneous sample, we aligned and profiled them against the original Sabin2 sequence. The profile revealed a large number of mutations and the two distinct relative frequencies created two baselines of mutations at the level of 10% and the second at the 30% (Fig S10a). After applying our algorithm with 1% mutation threshold we were able to reconstruct ten sequences with no mutations (Fig S10b)

Contigs less than 2000bp were ignored during all validation processes. For each experiment reconstructed sequences were extracted and aligned against all original sequences using hexagon[23]. Correct hits are reported as true positives (TP). False positives (FP) are the excess hits to the original sequences (anything that overlaps with the longest contig). False negatives (FN) are considered the original sequences with no hits. Furthermore, for true positives, the number of mismatches is reported as an additional measurement of accuracy.

The algorithm returns the depth of coverage per position and the frequency of each sequence was calculated based on the average depth of coverage of each haplotype. In order to measure the accuracy of the predicted relative frequencies, we use the Kullback-Leibler divergence between the predicted (P) frequency distribution and the true (T) distribution. Zero data points, which have resulted from false positives predictions, were disregarded.

We validated with using a sample of trivalent poliovirus vaccine made from a mixture of three serotypes of attenuated Sabin strains of poliovirus[34]. Sabin1, Sabin2, and Sabin3 sequences (Genbank accession numbers AY184219, AY184220, and AY184221 respectively) were used for Hexagon alignment with default parameters. This produced a total of 60∗106 alignments, which were then analyzed by the algorithm using 1% as mutation threshold. The three sequences were fully reconstructed and four more contigs of less than 150bp were also reported (Fig S15a). We predicted Sabin1 with relative frequency 93.58% and 17 mismatches, which correspond to 0.22% percent of the positions. Sabin2 was reconstructed with 0 mismatches despite the low frequency of 1.2%. Finally, the identified Sabin3 sequence had only 3 mismatched positions, an equivalent of 0.04% of the total length and 5.14% frequency (Fig S15b and Additional File 6). Very deep sequencing further supports the results because even for the lowest frequency Sabin2 virus the depth of coverage exceeds an 8000x. What appears as decrease in the depth of coverage in Fig.2C is due to the gaps introduced by the mutual alignment. The normalized representation of the graph highlights this effect, where relative frequency of a sequence increases locally to 100% in regions where these gaps occur.

**Datasets**

Artificial datasets were generated using a native HTS simulator. Sequence Sabin2 was used as a template to generate 10 more sequences, each representing a different population in the heterogeneous sample. Mutations were introduced at every 50 positions with a starting position such that no mutation was overlapping with another on a different sequence. Each of these sequences was used as a template to generate NGS reads with the characteristics included in Table S1. Artificially generated reads were merged into a single fastq file and aligned against the original Sabin2 sequence. Quality scores were all set to 30.

For the sensitivity analysis, six datasets were generated using Sabin2 as a template, consisting of 1 million short reads each. Initially two new sequences were created by introducing random mutations to Sabin2. Each sequence had 5% mutated position from the original sequence and approximately 9% differences from each other (Table S3). The two sequences were used to create these six samples by generating short reads of 250bp at different ratios with relative frequency ranging from 50% to 0.1% (Table S2).

In order to produce different levels of genetic distance, a set of 100 sequences were generate by randomly mutating Sabin2. The greater genetic distant was set to 50% and the space between the two most distant sequences was divided into steps of 0.5% genetic distance. The relative abundance of each of the 100 points was calculated using the power distribution.

Six datasets were generated each using an increasing number of sequences as templates chosen in such a way that each consecutive pair of references is equidistant with the previous one (Table S4). Distances between sequences for each sample are described in Tables S5-10.

**Comparison analysis**

The tools included in the comparison were QuRe\_v0.99971, ShoRah, ViSpa and PredictHaplo-1.0. Additionally ViQuaS1.3 and HaploClique were included in the comparisons but failed to run successfully in our environment and no results are reported. All tools were executed sequentially against all samples, with a specific time limit described in Table 3 and S12. All results were validated for the accuracy of the sequence reconstruction and abundance estimation. Trees were created from the sequences each tool predicted for each sample. Reconstructed sequences from each computation were aligned together with the original sequences of the corresponding sample using MUSCLE from MEGA7[35] with the default arguments. Trees were generated in MEGA7 using the neighbor-joining method[36] with default arguments. In order to associate the predicted sequences with the original ones, each tree was traversed using the B\* algorithm starting from one original sequence, looking for the first predicted sequence and removing the latter from the available leaves. The operation was repeated until all leaves of original sequences were matched to a leaf of predicted sequence or until no other leaves were available. False positives were considered either sequences that were not matched with original ones or that had mismatches to the associated original more than a given threshold. The trees were drawn using the ETE toolkit. In order to measure the accuracy of the predicted abundances, we used Jensen-Shannon divergence between the predicted (P) frequency distribution and the true (T) distribution.

where

The tools that successfully produced results for the samples SCL1-4 were validated by aligning the reconstructed sequences back to the original ones (Additional File 7). For those results where no predicted sequence aligned against any original, we performed a more thorough analysis using the genome comparator available in HIVE. 100.000 random reads were generated from the predicted sequences of each run and mixed with 100.000 random reads generated by the original sequences. The reads were mapped against both predicted and original sequences and chord graphs were generated. Reads aligned to two different sequences were used as evidence of the similarity between the regions of these sequences (Additional File 1 S21). Only Hexahedron was able to predict sequences associated to original sequences. Further validation was performed using the same techniques applied on the specificity analysis. Recall, precision, % of mismatches and Kullback-Leibler divergence were identical to the SP4 sample of the specificity analysis.

**Hexahedron workflow**

The implementation of the algorithm is part of HIVE platform that can be publicly accessed through this URL (<https://hive.biochemistry.gwu.edu/dna.cgi?cmd=main>). After logging in, using guest’s credential, the user may start by uploading the NGS reads through HIVE’s web interface. A set of tutorials that explain basic and more advance usage of HIVE is available too (<https://hive.biochemistry.gwu.edu/dna.cgi?cmd=tutorial>). Necessary input for the algorithm is an alignment of the sample against a number of references. Tutorial on performing alignment computations is also available through the same portal. Should multiple reference sequences have been selected for the alignment, an additional mutual alignment of the reference sequence is required before executing Hexahedron. A link to Hexahedron is available through the web page of the NGS sample alignment as an option of subsequent computations (“what’s next” section). Hexahedron webpage will display the alignment that will be used as an input and will request for the ID of multiple alignment process. Additional arguments include the mutation threshold, above which bifurcations are triggered, and a flag that will mark the HTS sample as pair-end reads.

**Architecture and computational environment**

All data displayed on the website as well as any data or references used in the analysis are stored in the High-performance Integrated Virtual Environment (HIVE) server (http://hive.biochemistry.gwu.edu). The results are accessible online and available for downloading. Users can use the algorithm as a next step after the alignment or browse the publicly available pre-existing results. HIVE provided computational infrastructure for storage and analysis for this project. Comparison of Hexahedron with other tools and performance tests were done on CentOS installed on an Intel 2 Quad core 2.26 GHz with 24GB of RAM system.

**Availability of supporting data**

The data sets supporting the results of this article are available in HIVE <https://hive.biochemistry.gwu.edu/review/Hexahedron%20publication>

**List of abbreviations**

HTS: High-throughput sequencing technologies

HIVE: High-performance Integrated Virtual Environment

SNV: Single Nucleotide Variations

TP: true positives

FP: false positives

FP: false negatives

KL: Kullback-Leibler (divergence)

JS: Jensen-Shannon divergence

PCR: polymerase chain reaction

HIV: Human Immunodeficiency Virus

**Competing interests**

The authors declare that they have no competing interests.

**Authors’ contribution**

KK conceived, designed developed, implemented, debugged, tested the algorithm and the visualization methods, conceived, designed, performed and collected the data of the *in silico* experiments and wrote the manuscript. VS participated in initial design and mentored in algorithmic concepts. KC participated in initial design provided valuable input on biological significance, evaluated the algorithm; provided biological sequences and analyzed using the tool. RM provided guidance to KK and was involved in streamlining the analysis method for routine analysis of high-throughput sequence data. All authors read, approved and participated in preparation of the final manuscript.

**Competing interests**

The authors declare that they have no competing interests.

**Additional files**

**Additional file 1: Includes the supplementary Figures S1–15.** **Fig S1** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 1 reference. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Increasing the allowed mismatches leads to improved results but still one reference is not enough to recover enough reads and assemble all sequences. **Fig S2** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against two references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Increasing the allowed mismatches leads to improved results but still one reference is not enough to recover enough reads and assemble all sequences. **Fig S3** Sankey diagrams of *in-silico* NGS reads from a mixed sample of 16 sequences aligned against three references. Each alignment was repeated 3 times changing the threshold of allowed mismatches to: 5% (**a**), 10% (**b**) and 15% (**c**). Increasing the allowed mismatches leads to improved results and we see that even two references are adequate to assemble all sequences with a loose alignment of 15% mismatches allowed. **Fig S4** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 5 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Five references are adequate to assemble all sequences even with alignment of 5% mismatches allowed. **Fig S5** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 10 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Ten references are adequate to assemble all sequences even with alignment of 5% mismatches allowed. **Fig S6** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 16 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Sixteen references are adequate to assemble all sequences even with alignment of 5% mismatches allowed. **Fig S7** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 50 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). The increased number of references does not have an impact on the assembly of the sequences. **Fig S8** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 100 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Saturating the reference space does not have an impact on the assembly of the sequences. **Fig S9** Sankey diagrams *in-silico* NGS reads from mixed sample of 16 sequences aligned against 500 references. Each alignment was repeated 3 times changing the threshold of allowed mismatches: 5% (**a**), 10% (**b**) and 15% (**c**). Oversaturation of the reference space does not have an impact on the assembly of the sequences. **Fig S10** Proof of concept assembly. **a**) Looking at a SNP profile only reveals two variants, one at 5% and another at 30%. **b**) Hexahedron resolved all haplotypes with 100% accuracy in sequence reconstruction and relative frequency detection. Depth of coverage for each sequence is represented by the width of each flow. **Fig S11** Sankeydiagrams of mixtures of two haplotypes in different concentrations. Mixtures of two sequences in 50:50, 90:10, 95:5, 99:1, 99.5:0.5 and 99.9:0.1 proportion were analyzed and results are presented in form of Sankey diagrams **a**, **b**, **c**, **d**, **e** and **f** respectively. **Fig S12** Sankey diagrams of mixtures of 2,4,8,16,32 and 64 sequences. Results are presented in **a**, **b**, **c**, **d**, **e** and **f** respectively in form of Sankey diagrams. Samples with number of sequences up to 16(**a**, **b**, **c** and **d**) are resolved in a global scale. Sample with higher number of sequences (**e** and **f**) are not fully resolved resulting in bifurcations along the genome and smaller contigs. **Fig S13** Chord diagrams of mixtures of 2,4,8,16,32 and 64 sequences. Chord diagrams represent the similarities of assembled against original sequence for the mixture of 2, 4, 8, 16, 32 and 64 samples (**a**, **b**, **c**, **d**, **e**, and **f**). **a-d)** there is a unique association between all identified sequence and all original sequences. **e** and **f)** Although there are few many to many relations, most of the sequences are also uniquely linked to one original sequence. **Fig S14** Hexahedron speed performance. Noise affects the computation time indirectly, only in computation with distance between sequences smaller than the noise. This would cause more bifurcations leading to a linear decrease of speed. The number of sequences in the sample had a linear correlation with time of computation and the distance within the sequences affected the speed, where the more closely related sequences exhibiting slower speeds. Finally, the number of reads in the sample had a sub-linear correlation with the speed. As described by the complexity of the algorithm, this effect was amplified by the increasing number of sequences in the sample. **Fig S15** Reconstruction of oral poliovirus vaccine. **a**) A sample of the trivalent poliovirus vaccine with known sequences and frequency distribution. Only three haplotypes reconstructed longer than 150bp each exhibiting similarity exclusively to one of the Sabin references. **b**) Relative frequency of each sequence detected in the trivalent vaccine. **Figure S16** Precision results for Hexahedron, PredictHaplo, QuRe, ShoRah and ViSpA obtained for 20 *in silico* samples containing of 4 populations, each at 25% abundance. Each column of charts represents a different number of short reads in the samples and each row a different sequence length of the populations. For each computation TP are called based on the number of mismatches between the reconstructed and the original sequences. Precision is reported based on mismatch thresholds 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5%. Missing data points are due to the failure to obtain results within the period specified for each sample. **Figure S17** Recall results for Hexahedron, PredictHaplo, QuRe, ShoRah and ViSpA. Each column of charts represents a different number of short reads in the samples and each row a different sequence length of the populations. For each computation True Positives (TP) are called based on the number of mismatches between the reconstructed and the original sequences. Recall is reported based on mismatch thresholds 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5%. Missing data points are due to the failure to obtain results within the period specified for each sample. **Figure S18** False Positive (FP) results for Hexahedron, PredictHaplo, QuRe, ShoRah and ViSpA. Each column of charts represents a different number of short reads in the samples and each row a different sequence length of the populations. FP are reported based on mismatch thresholds 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5% between the reconstructed and the original sequences. Missing data points are due to the inability failure to obtain results within the period specified for each sample. **Figure S19** False Negative (FN) results for Hexahedron, PredictHaplo, QuRe, ShoRah and ViSpA. Each column of charts represents a different number of short reads in the samples and each row a different sequence length of the populations. FN are reported based on mismatch thresholds 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5% between the reconstructed and the original sequences. Missing data points are due to the failure to obtain results within the period specified for each sample. **Figure S20** Neighbor-joining trees of original and reconstructed sequences from all tools. ShoRah leaves are colored red, PredictHaplo green, Hexahedron blue, QuRe yellow, ViSpa light blue and original sequences are colored black. Blue circles next to leaves represent the abundance predicted for the corresponding sequence. Tree of sequences generated from samples SCS1-4 (**a-d**), SCS6-10 (**e-i**) and SCS12-20 (**j-q**). **Figure S21** Chord diagrams of PredictHaplo reconstructed sequences. Chord diagrams represent the similarities of PredictHaplo reconstructed against original sequence for the 100K, 500K and 1M samples (**a**, **b** and **c**).

**Additional file 2: Includes the Supplementary Tables S1-5 Table S1.** Artificially generated NGS reads for heterogeneous sample. **Table S2.** Artificially generated NGS reads of two populations for sensitivity test. **Table S3.** Distance between sequences in SN1 calculated in MEGA7. **Table S4.** Artificially generated NGS reads of populations for specificity test. **Table S5.** Distance between sequences in SP1 calculated in MEGA7. **Table S6.** Distance between sequences in SP2 calculated in MEGA7. **Table S7.** Distance between sequences in SP3 calculated in MEGA7. **Table S8.** Distance between sequences in SP4 calculated in MEGA7. All ambiguous positions were removed for each sequence pair. **Table S9.** Distance between sequences in SP5 calculated in MEGA7. **Table S10.** Distance between sequences in SP6 calculated in MEGA7. **Table S11.** Distance between sequences in samples SCS1-20 calculated in MEGA7. **Table S12.** Artificially generated NGS reads of 16 populations. Abundance follows the power distribution and all tools were given a specific period to perform the computations depending on the size of each sample. **Table S13** Summary of results from Hexahedron, QuRe, PredictHaplo, ViSpa and ShoRah against samples SCL1-4. Only PredictHaplo gave results within a limited period but not for a NGS sample of 5 million reads. Reconstructed sequences were then aligned back to the original sequence in order to associate each predicted sequence to the original one.

**Additional file 3:** Includes the reconstructed sequences produced by the benchmarked tools.

**Additional file 4:** Includes a summary of the reconstructed sequences, the letter sequences and a per position depth of coverage for each sequence produced from the sensitivity experiment.

**Additional file 5:** Includes a summary of the reconstructed sequences, the letter sequences and a per position depth of coverage for each sequence produced from the specificity experiment.

**Additional file 6:** Includes a summary of the reconstructed sequences, the letter sequences and a per position depth of coverage for each sequence produced from all real sample runs.

**Additional file 7:** Validation of Hexahedron and PredictHaplo results on benchmark samples. Resulted sequences from 100K, 500K,1M and 5M (only for Hexahedron) read samples were aligned back to the original sequences and the number of mismatches, insertions and deletions are reported.

**Acknowledgments**

We thank M. Laassri who generated the deep sequence reads for Mumps and Sabin vaccine and S. Rubin who provided us with the mumps sample. Konstantinos Karagiannis is a predoctoral student in the Biochemistry and Systems Biology Program of the Institute for the Biomedical Sciences at the George Washington University. This work is from a dissertation to be presented to the above program in partial fulfillment of the requirements for the Ph.D. degree.

REFERENCES

1. Sanjuan R, Nebot MR, Chirico N, Mansky LM, Belshaw R: **Viral mutation rates.** *J Virol* 2010, **84:**9733-9748.

2. Holmes EC: **The RNA virus quasispecies: fact or fiction?** *J Mol Biol* 2010, **400:**271-273.

3. Arenas M, Lorenzo-Redondo R, Lopez-Galindez C: **Influence of mutation and recombination on HIV-1 in vitro fitness recovery.** *Mol Phylogenet Evol* 2015.

4. Lundgren JD, Babiker AG, Gordin FM, Borges AH, Neaton JD: **When to start antiretroviral therapy: the need for an evidence base during early HIV infection.** *BMC Med* 2013, **11:**148.

5. Korber B, Gaschen B, Yusim K, Thakallapally R, Kesmir C, Detours V: **Evolutionary and immunological implications of contemporary HIV-1 variation.** *Br Med Bull* 2001, **58:**19-42.

6. Rossi LM, Escobar-Gutierrez A, Rahal P: **Advanced molecular surveillance of hepatitis C virus.** *Viruses* 2015, **7:**1153-1188.

7. Sanger F, Coulson AR: **A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase.** *J Mol Biol* 1975, **94:**441-448.

8. Eriksson N, Pachter L, Mitsuya Y, Rhee SY, Wang C, Gharizadeh B, Ronaghi M, Shafer RW, Beerenwinkel N: **Viral population estimation using pyrosequencing.** *PLoS Computational Biology* 2008, **4:**e1000074.

9. Willerth SM, Pedro HA, Pachter L, Humeau LM, Arkin AP, Schaffer DV: **Development of a low bias method for characterizing viral populations using next generation sequencing technology.** *PLoS ONE* 2010, **5:**e13564.

10. Metzker ML: **Sequencing technologies - the next generation.** *Nat Rev Genet* 2010, **11:**31-46.

11. Zagordi O, Daumer M, Beisel C, Beerenwinkel N: **Read length versus depth of coverage for viral quasispecies reconstruction.** *PLoS ONE* 2012, **7:**e47046.

12. Prosperi MC, Yin L, Nolan DJ, Lowe AD, Goodenow MM, Salemi M: **Empirical validation of viral quasispecies assembly algorithms: state-of-the-art and challenges.** *Sci Rep* 2013, **3:**2837.

13. Yang X, Charlebois P, Macalalad A, Henn MR, Zody MC: **V-Phaser 2: variant inference for viral populations.** *BMC Genomics* 2013, **14:**674.

14. Mangul S, Wu NC, Mancuso N, Zelikovsky A, Sun R, Eskin E: **Accurate viral population assembly from ultra-deep sequencing data.** *Bioinformatics* 2014, **30:**i329-337.

15. Schirmer M, Sloan WT, Quince C: **Benchmarking of viral haplotype reconstruction programmes: an overview of the capacities and limitations of currently available programmes.** *Brief Bioinform* 2014, **15:**431-442.

16. Yang X, Charlebois P, Gnerre S, Coole MG, Lennon NJ, Levin JZ, Qu J, Ryan EM, Zody MC, Henn MR: **De novo assembly of highly diverse viral populations.** *BMC Genomics* 2012, **13:**475.

17. Topfer A, Marschall T, Bull RA, Luciani F, Schonhuth A, Beerenwinkel N: **Viral quasispecies assembly via maximal clique enumeration.** *PLoS Computational Biology* 2014, **10:**e1003515.

18. Prosperi MC, Salemi M: **QuRe: software for viral quasispecies reconstruction from next-generation sequencing data.** *Bioinformatics* 2012, **28:**132-133.

19. Prosperi MC, Prosperi L, Bruselles A, Abbate I, Rozera G, Vincenti D, Solmone MC, Capobianchi MR, Ulivi G: **Combinatorial analysis and algorithms for quasispecies reconstruction using next-generation sequencing.** *BMC Bioinformatics* 2011, **12:**5.

20. Beerenwinkel N, Zagordi O: **Ultra-deep sequencing for the analysis of viral populations.** *Curr Opin Virol* 2011, **1:**413-418.

21. Astrovskaya I, Tork B, Mangul S, Westbrooks K, Mandoiu I, Balfe P, Zelikovsky A: **Inferring viral quasispecies spectra from 454 pyrosequencing reads.** *BMC Bioinformatics* 2011, **12 Suppl 6:**S1.

22. Zagordi O, Klein R, Daumer M, Beerenwinkel N: **Error correction of next-generation sequencing data and reliable estimation of HIV quasispecies.** *Nucleic Acids Res* 2010, **38:**7400-7409.

23. Santana-Quintero L, Dingerdissen H, Thierry-Mieg J, Mazumder R, Simonyan V: **HIVE-hexagon: high-performance, parallelized sequence alignment for next-generation sequencing data analysis.** *PLoS ONE* 2014, **9:**e99033.

24. Kullback S: **The Kullback-Leibler Distance.** *American Statistician* 1987, **41:**340-340.

25. Rezapkin GV, Fan L, Asher DM, Fibi MR, Dragunsky EM, Chumakov KM: **Mutations in Sabin 2 strain of poliovirus and stability of attenuation phenotype.** *Virology* 1999, **258:**152-160.

26. Katoh K, Standley DM: **MAFFT multiple sequence alignment software version 7: improvements in performance and usability.** *Mol Biol Evol* 2013, **30:**772-780.

27. Afzal MA, Pickford AR, Forsey T, Heath AB, Minor PD: **The Jeryl Lynn vaccine strain of mumps virus is a mixture of two distinct isolates.** *J Gen Virol* 1993, **74 ( Pt 5):**917-920.

28. Zagordi O, Bhattacharya A, Eriksson N, Beerenwinkel N: **ShoRAH: estimating the genetic diversity of a mixed sample from next-generation sequencing data.** *BMC Bioinformatics* 2011, **12:**119.

29. Prabhakaran S, Rey M, Zagordi O, Beerenwinkel N, Roth V: **HIV Haplotype Inference Using a Propagating Dirichlet Process Mixture Model.** *IEEE/ACM Trans Comput Biol Bioinform* 2014, **11:**182-191.

30. Prabhakaran S, Rey M, Zagordi O, Beerenwinkel N, Roth V: **HIV Haplotype Inference Using a Propagating Dirichlet Process Mixture Model.** *IEEE/ACM Trans Comput Biol Bioinform* 2013.

31. Behera AK, Basu S, Cherian SS: **Molecular mechanism of the enhanced viral fitness contributed by secondary mutations in the hemagglutinin protein of oseltamivir resistant H1N1 influenza viruses: modeling studies of antibody and receptor binding.** *Gene* 2015, **557:**19-27.

32. Pfeiffer JK, Kirkegaard K: **A single mutation in poliovirus RNA-dependent RNA polymerase confers resistance to mutagenic nucleotide analogs via increased fidelity.** *Proceedings of the National Academy of Sciences of the United States of America* 2003, **100:**7289-7294.

33. Lange CM, Hue S, Violari A, Cotton M, Gibb D, Babiker A, Otwombe K, Panchia R, Dobbels E, Jean-Philippe P, et al: **Single Genome Analysis for the Detection of Linked Multiclass Drug Resistance Mutations in HIV-1-Infected Children After Failure of Protease Inhibitor-Based First-Line Therapy.** *J Acquir Immune Defic Syndr* 2015, **69:**138-144.

34. Sabin AB, Ramos-Alvarez M, Alvarez-Amezquita J, Pelon W, Michaels RH, Spigland I, Koch MA, Barnes JM, Rhim JS: **Live, orally given poliovirus vaccine. Effects of rapid mass immunization on population under conditions of massive enteric infection with other viruses.** *JAMA* 1960, **173:**1521-1526.

35. Kumar S, Stecher G, Tamura K: **MEGA7: Molecular Evolutionary Genetics Analysis Version 7.0 for Bigger Datasets.** *Mol Biol Evol* 2016, **33:**1870-1874.

36. Saitou N, Nei M: **The neighbor-joining method: a new method for reconstructing phylogenetic trees.** *Mol Biol Evol* 1987, **4:**406-425.

**Figure legends**

**Fig. 1** Hexahedron algorithm overview. The algorithm creates bifurcations in every point mutation that exceeds a user specified threshold and maintains the contig as long as possible. It takes advantage of the existing alignment data and correlates distant mutations, reported by an alignment algorithm possibly because of the selection of a distant reference genome. Hexahedron makes no statistical assumption; it rather extracts the information from the overlapping alignments in a step wise, contig assembly, fashion, using coordinates of the aligned reads mapped to the mutual alignment of the references.

**Fig. 2** Populations assembly results of real sample. **a**) A mixture of mumps was resolved into exactly two sequence with a frequencies of 94% and 6%(**b**). **c**) Sample isolated from concentrated sewage water in RD cell culture. Only two sequences longer than 1000bp were reconstructed. The two sequences were identical in the 3’ end indicated by the shorter reconstructed merged to the dominant one at position 4325. **d**) Relative frequency of the two sequences at 88.67% and 11.23%. **e**) Another sample isolated from sewage concentrate using RD cell culture was analyzed using our software. A more complex picture reveals 3 fully reconstructed sequence and a number of smaller contigs that depict the dynamic nature of the virus. **f**) Frequency distribution of identified populations in the environmental sample.

**Fig. 3** Comparison studies results. Hexahedron, PredictHaplo, QuRe, ShoRah and ViSpA were compared using 20 *in silico* samples of 4 populations, each one of 25% abundance. **a**) F-scores. Each column of charts represents a different number of short reads in the samples and each row a different sequence length of the populations. For each computation True Positive(TP) are called based on the number of mismatches between the reconstructed and the original sequences. F-scores are reported based on mismatch thresholds 0%, 0.01%, 0.05%, 0.1%, 0.5%, 1% and 5%.Missing data points are due to the failure to obtain results within the period specified for each sample. **b**) Jensen-Shannon divergence between the predicted and the original frequencies. **c**) Speed performance of the tools reported in minutes. The results indicate an time increases exponential as the number of short reads increases for PredictHaplo, QuRe, ShoRah and ViSpA.

**Fig. 4** Neighbor-joining trees of original and reconstructed sequences from all tools. ShoRah leaves are colored red, PredictHaplo green, Hexahedron blue, QuRe yellow, ViSpa light blue and original sequences are colored black. Blue circles next to leaves represent the abundance predicted for the corresponding sequence. **a**) Tree of sequences generated from sample SCS5. Two of the PredictHaplo reconstructed sequences are assigned to nodes closer to their original sequences compared to Hexahedron but PredictHaplo has reconstructed 5 sequences instead of five. ShoRah did not conclude the computation within the given time frame. **b**) Sample SCS11. QuRe did not produce results.

**Table 1** Impact of increased number of references on sequence assembly.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **# of References** | **Mismatches allowed (%)** | **Unaligned reads (%)** | **Precision (%)** | **Recall (%)** |
| 1 | 5 | 79.5 | 100 | 37.5 |
| 1 | 15 | 25 | 100 | 87.5 |
| 3 | 5 | 24.8 | 100 | 87.5 |
| 3 | 15 | 0.1 | 100 | 100 |
| 10 | 5 | 0 | 100 | 100 |
| 500 | 15 | 0 | 100 | 100 |

**Table 2** Sequence reconstruction accuracy and Kullback-Leibler divergence between known and predicted frequency distribution.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample** | **Recall (%)** | **Precision (%)** | **Av. Mismatches (%)** | **KL-Divergence** |
| **Sensitivity** | | | | |
| SN1 | 100 | 100 | 0.0067 | 6.8 10-6 |
| SN2 | 100 | 100 | 0.0067 | 0.00043 |
| SN3 | 100 | 100 | 0.0067 | 0.00024 |
| SN4 | 100 | 100 | 0.0067 | 5.6 10-5 |
| SN5 | 100 | 100 | 0.0067 | 3.2 10-5 |
| SN6 | 100 | 100 | 0.0067 | 5.7 10-5 |
| **Specificity** | | | | |
| SP1 | 100 | 100 | 0.0605 | 4.5 10-06 |
| SP2 | 100 | 100 | 0 | 3.4 10-06 |
| SP3 | 100 | 100 | 0.1519 | 4.8 10-05 |
| SP4 | 100 | 100 | 0.0439 | 2.5 10-5 |
| SP5 | 84.375 | 84.375 | 0.6934 | 0.00786 |
| SP6 | 60.9375 | 50.64935 | 1.4272 | 0.03951 |

**Table 3** Summary of *in silico* samples generated for comparison studies. Samples with the SCS prefix have four haplotypes with uniform abundance distribution and samples with SCL prefix have 16 haplotypes with power distribution. Column “WC limit” describes the Wall Clock time, in hours, given to tools to run each sample.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Length (bp)** | **# of short reads** | **Av. Depth/haplotype** | **WC limit (h)** | **# of haplotypes** |
| SCS1 | 500 | 1000 | 125 | 1 | 4 |
| SCS2 | 500 | 5000 | 625 | 2 | 4 |
| SCS3 | 500 | 10000 | 1250 | 3 | 4 |
| SCS4 | 500 | 50000 | 6250 | 6 | 4 |
| SCS5 | 500 | 100000 | 12500 | 12 | 4 |
| SCS6 | 1000 | 1000 | 62.5 | 1 | 4 |
| SCS7 | 1000 | 5000 | 312.5 | 2 | 4 |
| SCS8 | 1000 | 10000 | 625 | 3 | 4 |
| SCS9 | 1000 | 50000 | 3125 | 6 | 4 |
| SCS10 | 1000 | 100000 | 6250 | 12 | 4 |
| SCS11 | 2000 | 1000 | 31.25 | 1 | 4 |
| SCS12 | 2000 | 5000 | 156.25 | 2 | 4 |
| SCS13 | 2000 | 10000 | 312.5 | 3 | 4 |
| SCS14 | 2000 | 50000 | 1562.5 | 6 | 4 |
| SCS15 | 2000 | 100000 | 3125 | 12 | 4 |
| SCS16 | 5000 | 1000 | 12.5 | 1 | 4 |
| SCS17 | 5000 | 5000 | 62.5 | 2 | 4 |
| SCS18 | 5000 | 10000 | 125 | 3 | 4 |
| SCS19 | 5000 | 50000 | 625 | 6 | 4 |
| SCS20 | 5000 | 100000 | 1250 | 12 | 4 |
| SCL1 | ~7200 | 100000 | 2417 - 926937 | 12 | 16 |
| SCL2 | ~7200 | 500000 | 483.5 - 185387.5 | 24 | 16 |
| SCL3 | ~7200 | 1000000 | 241.7 - 92693.7 | 48 | 16 |
| SCL4 | ~7200 | 5000000 | 48.3 - 18538.8 | 96 | 16 |