Supplementary Materials and Methods

Scripts to run VirGA can be downloaded from https://bitbucket.org/szparalab, and documentation for their use is found at http://virga.readthedocs.org/. VirGA outputs from this publication are archived in a repository at https://scholarsphere.psu.edu/collections/sf268c193. This repository also provides offers the option of downloading a virtual machine (VM) which contains a full installation of VirGA.

VirGA Step 1 – Raw Read Preprocessing

For *de novo* assembly to be successful, high-quality and contaminant-free sequencing reads must be used. VirGA's first step analyzes the raw sequence reads using FastX-Toolkit (http://hannonlab.cshl.edu/fastx_toolkit/) and FASTQC (http://www.bioinformatics.babraham.ac.uk/projects/fastqc/) to generate histograms of base quality scores and other relevant metrics. The reads are then trimmed of any adapters or sequencing artifacts via FastX-Toolkit and low quality bases are trimmed using a sliding window algorithm in Trimmomatic (1). Length filtering removes any reads that fall below a specified threshold. Next, host contamination is removed by using Bowtie 2 (2) to identify any sequences matching the host genome. For this analysis, we use the rhesus macaque genome as a proxy for the African green monkey (Vero) cells used to grow these viruses. Reads left unpaired as a result of the above filtering steps are likewise removed. Finally, FastX-Toolkit and FASTQC are again used to generate comparable post-processing metrics for the read files.

VirGA Step 2 – *de novo* Contig Assembly

The cleaned sequence read data are then used as the input for multiple different permutations of SSAKE (3) *de novo* assembly. Each assembly uses a different set of node overlap (-m) and trimming (-t) parameters, which have been optimized for reads between 100 to 300 bases long. The default settings use eight (8) pairs of overlap and trimming parameters: 16-0, 19-0, 20-0, 29-0, 16-4, 19-4, 20-4, 29-4. Advanced users may specify any number of overlap and trimming parameter pairs. All SSAKE assembly jobs are orchestrated by the central VirGA workflow and are therefore compatible with PBS or Torque scheduling systems and can be conducted in parallel.

At the conclusion of the SSAKE assemblies, the numerous contigs produced are collected into a file and used as input for the Celera *de novo* assembler (4). VirGA treats these contigs as long reads, for which quality scores are required. Custom quality scores are generated for each contig, which allows the user to specify various per-base confidence distributions with greater weight typically given to proximal bases. Celera combines many overlapping contigs and joins some contigs to provide a set of contigs with much less repetition and a better N50.

VirGA Step 3 – Genome Linearization and Annotation

To combine the Celera contigs into a draft genome, VirGA uses a completed reference genome of a closely related strain as a guide. In this case, we used the HSV1 strain 17 (NCBI Reference Sequence: JN_555585.1). The Mugsy whole genome alignment software (5) (http://mugsy.sourceforge.net/) is used to align the Celera contigs to the reference and the resulting synteny blocks are stitched together using a custom script, called maf_net.py, from the VAMP toolkit. This results in a linear, reference-guided organization of *de novo* assembled contig sequences. If desired, the program GapFiller (6) can then implemented for focused local reassembly at the regions of the genome containing gaps. This often results in gap closure and a more robust genome. When GapFiller is used, VirGA implements a second round of Mugsy and maf_net, to generate an appropriate alignment file for downstream annotation.

Once a draft genome is generated, VirGA then annotates and compare the new draft genome to the reference. To do this, VirGA uses the Mugsy alignment of the new draft genome to the reference genome, and a script called compare_genomes.py from the VAMP toolkit. This step uses positional homology to locate the boundaries of all features previously identified in the reference genome, and transfer them to the new draft genome. These features, including both coding and non-coding sequences, are recorded in a GFF3 formatted annotation file.

VirGA Step 4 – Assembly assessment

Using the fully annotated draft genome assembly, VirGA's fourth and final step checks the quality of the assembly using several strategies. First, the preprocessed reads are mapped back to the draft genome using Bowtie 2, followed by SAMTools (7) to generate coverage and pileup information. SAMTools and FreeBayes (8) are then used to detect variants, or polymorphisms, and to

determine the abundance of the reference vs. alternate bases. Because herpesvirus populations can be polymorphic, VirGA checks that the final consensus genome reflects the majority base at any site with evidence of sequence polymorphism. VirGA requires polymorphisms to meet the following criteria: a minimum depth of 80 sequence reads, a maximum strand bias of 80%, and no more than 5 homopolymer bases flanking the variant. For any polymorphic site meeting these criteria, VirGA compares the quantity of sequence support for the currently called base versus any alternate base. If the alternate base has greater support than the base included the draft consensus genome, the genome is corrected to reflect the majority base. After all variant corrections are complete, VirGA repeats the alignment and annotation steps (Mugsy, maf_net, and compare_genomes) to generate a final consensus genome and annotation file. The preprocessed reads are mapped to this final consensus genome again, so that the final alignment file contains accurate positional data. Variants are then detected again using SAMTools and Freebayes, to produce two variant summary files that record the location, alternative base(s), and sequence support for any polymorphic sites. Several custom scripts manipulate the pileup file in conjunction with final assembly and annotation files to produce a graphical representation of assembly, including per-base coverage, all coding and non-coding features, and to highlight areas of low or no coverage (Fig. S4).

The annotated genes of the final assembly are thoroughly inspected to ensure quality. Each gene, including individual exons, is compared to the homologous reference gene through ClustalW[2] (9) alignment at the DNA, transcript, and amino acid levels. To pass inspection, a particular gene must have a start and stop codon, no gaps, no less than 80% sequence identity to the reference, and must share a stop codon in a similar position as the reference. Genes failing to meet these criteria are flagged for user inspection (Fig. S4).

At the conclusion of VirGA's fourth step, a detailed HTML report is generated with embedded hyperlinks to all files concerning the genomes, annotations and alignments (Fig. S4). VirGA also includes an option (on by default) to generate a full-length genome for GenBank submission, from the trimmed format generated by *de novo* assembly (see Figure 3 for example). Terminal repeats are generated by inverting the internal structural repeats (IRL and IRS) and appending these sequences to the genome boundaries (10). Detailed metrics from each step as well as numerous graphics are also included.

Genetic distance analysis

ClustalW was used to align the trimmed genome sequences of all HSV-1 strains in Table 3, along with several strains from NCBI for comparison (accessions listed in Fig, S6 legend). Genetic distance was calculated using the unweighted-
pair group method with arithmetic mean (UPGMA) in MEGA (11), with 1,000 bootstrap replicates and the maximum composite likelihood method for distance estimation (12).

Restriction fragment length polymorphism analysis

Restriction fragment length polymorphism (RFLP) analysis of sub-clones of HSV-1 KOS and F were done using BamHI and HindIII (New England BioLabs; NEB). Each 30 µL digest contained 0.5 µg viral nucleocapsid DNA, either 40 units HindIII or 200 units BamHI enzyme, and 2 µL 10X Buffer (NEB). Digests were run overnight, and then fragments were separated via overnight electrophoresis on 0.5% agarose gels, and visualized after stained with ethidium bromide. Migration markers included a 1 kb ladder and HindIII Lambda DNA ladder (both from NEB).

References

1. **Bolger AM, Lohse M, Usadel B**. 2014. Trimmomatic: a flexible trimmer for Illumina sequence data. Bioinformatics **30**:2114–2120.
2. **Langmead B, Salzberg SL**. 2012. Fast gapped-read alignment with Bowtie 2. Nat. Methods **9**:357–359.
3. **Warren RL, Sutton GG, Jones SJM, Holt R a**. 2007. Assembling millions of short DNA sequences using SSAKE. Bioinformatics **23**:500–1.
4. **Myers EW, Sutton G, Delcher AL, Dew IM, Fasulo DP, Flanigan MJ, Kravitz SA, Mobarry CM, Reinert KHJ, Remington KA, Anson EL, Bolanos RA, Chou H-H, Jordan CM, Halpern AL, Lonardi S, Beasley EM, Brandon RC, Chen L, Dunn PJ, Lai Z, Liang Y, Nusskern DR, Zhan M, Zhang Q, Zheng X, Rubin GM, Adams MD, Venter CJ**. 2000. A Whole-Genome Assembly of Drosophila. Science **287**:2196–2204.
5. **Angiuoli SV, Salzberg SL**. 2011. Mugsy: fast multiple alignment of closely related whole genomes. Bioinformatics **27**:334–342.
6. **Boetzer M, Pirovano W**. 2012. Toward almost closed genomes with GapFiller. Genome Biol. **13**:1–9.
7. **Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R, 1000 Genome Project Data Processing Subgroup**. 2009. The Sequence Alignment/Map format and SAMtools. Bioinformatics **25**:2078–2079.

8. **Garrison E, Marth G**. 2012. Haplotype-based variant detection from short-read sequencing. ArXiv Prepr. ArXiv12073907.
9. **Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, Higgins DG**. 2007. Clustal W and Clustal X version 2.0. Bioinformatics **23**:2947–2948.
10. **Szpara ML, Gatherer D, Ochoa A, Greenbaum B, Dolan A, Bowden RJ, Enquist LW, Legendre M, Davison AJ**. 2014. Evolution and diversity in human herpes simplex virus genomes. J. Virol. **88**:1209–27.
11. **Kumar S, Tamura K, Nei M**. 1994. MEGA: Molecular Evolutionary Genetics Analysis software for microcomputers. Comput. Appl. Biosci. CABIOS **10**:189–191.
12. **Tamura K, Nei M, Kumar S**. 2004. Prospects for inferring very large phylogenies by using the neighbor-joining method. Proc. Natl. Acad. Sci. U. S. A. **101**:11030–11035.