PHAST: A Fast Phage Search Tool

You Zhou[1], Yongjie Liang[2], Karlene H. Lynch[1], Jonathan J. Dennis[1] and David S. Wishart[1,2,3,*]

[1]Department of Biological Sciences, [2]Department of Computing Science, University of Alberta and [3]National Research Council, National Institute for Nanotechnology (NINT), Edmonton, AB, Canada T6G 2E8

Received February 28, 2011; Revised April 21, 2011; Accepted May 26, 2011

ABSTRACT

PHAge Search Tool (PHAST) is a web server designed to rapidly and accurately identify, annotate and graphically display prophage sequences within bacterial genomes or plasmids. It accepts either raw DNA sequence data or partially annotated GenBank formatted data and rapidly performs a number of database comparisons as well as phage 'cornerstone' feature identification steps to locate, annotate and display prophage sequences and prophage features. Relative to other prophage identification tools, PHAST is up to 40 times faster and up to 15% more sensitive. It is also able to process and annotate both raw DNA sequence data and Genbank files, provide richly annotated tables on prophage features and prophage 'quality' and distinguish between intact and incomplete prophage. PHAST also generates downloadable, high quality, interactive graphics that display all identified prophage components in both circular and linear genomic views. PHAST is available at (http://phast.wishartlab.com).

INTRODUCTION

Bacteriophage, the viruses that infect bacteria, can typically be divided into two groups, lytic and temperate. Lytic phage infect propagate within and then lyse their host bacterial cells as part of their life cycle, while temperate phages may exist benignly within the DNA of their bacterial host. Temperate phages can physically integrate into one of the native replicons (plasmid or chromosome) of their preferred bacterial host, although a few phages can exist as independent plasmids (1). Integrated phages are termed prophages. Prophages tend to be inserted at specific integration sites within the host genome, but their location can vary depending on the phage species. Prophages are essentially dormant phages that are only replicated through bacterial DNA replication and cell division. Bacteria containing a prophage are called lysogens because their prophage is in the lysogenic cycle, in which the viral (esp. the lytic) genes are not expressed. Upon damage to the host cell DNA or other physiological cues, the prophage may be induced to excise itself from the bacterial genome. After induction, the phage's lytic genes are turned on, infectious virions are assembled within the host cell and the cell is lysed (killed) releasing infectious phage particles that can go on to infect more cells.

Bacterial genomes can contain a significant proportion (>20%) of functional and non-functional bacteriophage genes (1). Consequently, prophage sequences can account for a significant fraction of the variation within bacterial species or clades. The presence of prophage sequences may also allow some bacteria to acquire antibiotic resistance, to exist in new environmental niches, to improve adhesion or to become pathogenic (1). Because bacterial genome fragments can also be carried by phage particles, the lytic process is thought to be an important vehicle for horizontal gene transfer. Furthermore, because of their high specificity and high potency, phages are also being investigated as potential candidates for novel antibiotics (2) or even cancer therapies (3). As the most abundant biological entity on Earth (1), phages are also thought to play a crucial role in cycling nutrients and boosting photosynthesis in the world's oceans (4). However, not all prophages or prophage-like entities are functional. Indeed, many prophages are dormant due to mutational decay or the loss of critical genes over thousands of host generations. Defective or cryptic prophages are abundant in many bacterial genomes and they can carry a number of genes that may be beneficial to the host. These can include genes encoding proteins with homologous recombination functions or bacteriocins that may be used to inhibit the growth of other bacteria in competition for nutrients.

There are generally two methods to identify prophages: (i) experimental and (ii) computational. The experimental approach involves inducing the host bacteria to release phage particles by exposing them to UV light or other DNA-damaging conditions. This approach can certainly prove the existence of viable phages, but will not reveal defective prophages (1). In addition, not all viable phages

[*] To whom correspondence should be addressed. Tel: +780 492 0383; Fax: +780 492 5305; Email: david.wishart@ualberta.ca

can be induced under the same conditions and the required conditions often are not known *a priori*. Given the ease with which bacterial genomes can now be sequenced, the computational identification of prophages from genomic sequence data has become the most preferred route. Early sequence-based efforts often depended upon manual inspection of disrupted genes and attachment sites (5) or the analysis of atypical nucleotide content (6,7). However, prophage regions do not always exhibit atypical nucleotide content (8). Likewise, phages neither always integrate into the same coding regions nor do they exclusively use tRNAs as the target site for integration. Consequently, this makes scanning for atypical gene content or simple searches for disrupted genes or tRNAs unreliable for finding prophage regions. More recent methods relying on a much more holistic or integrated approach have appeared. These combine sequence comparisons to known phage or prophage genes, comparisons to known bacterial genes, tRNA and dinucleotide analysis and hidden Markov scanning for attachment site recognition. These combined methods are now available in a number of excellent programs and web servers such as Phage_Finder (9), Prophinder (10) and Prophage Finder (11). These tools have helped revolutionize finding prophages in bacterial genomes.

Except for Prophage Finder, these phage finding programs and web servers still require that the input genome sequence must be well annotated with all open reading frames (ORFs) and/or tRNA sites pre-identified. This annotation process is not only time consuming, it is also highly dependent on the choice of the annotation programs or methods. Furthermore, the choice and accuracy of the genome annotation method can significantly affect the accuracy of the phage predictions (*vide infra*). In addition, even with a fully annotated bacterial genome, all other phage-finding methods require 30 min to 2 h to complete their analyses. Now that it is possible to sequence an entire bacterial genome in less than a day, prophage identification needs to be faster, more accurate and much less dependent on the availability of fully annotated bacterial genomes. To address these issues, we have developed a web-based application named PHAST (a fast PHAge Search Tool), to support rapid and accurate prophage identification using either raw or annotated bacterial genome sequence data. The main features of PHAST include:

- Prophage region identification support for both raw nucleotide sequence input (using GLIMMER gene prediction and local genome annotation tools) as well as annotated GenBank file input;
- Support for detailed prophage annotation including position, length, boundaries, number of genes, attachment sites, tRNAs, identified phage-like genes and attachment sites (*att*);
- A customized phage and prophage database that is automatically updated on a biweekly basis;
- Support for the prediction of the completeness or potential viability of identified prophages (intact, questionable or incomplete);
- Extremely fast processing times (about 3 min for a typical bacterial genome);
- Graphical output that supports both circular and linear genomic views as well as interactive browsing and labeling of dynamically generated figures;
- Fully downloadable text and graphics; and
- Support for scriptable operations through an application programming interface (API).

PHAST's prophage finding performance is generally superior to other applications. When given an annotated genome, it achieves 85.4% sensitivity and 94.2% positive predictive value (PPV) when evaluated against the collection of prophages referenced by Prophinder. When given a raw sequence file, PHAST achieves 79.4% sensitivity and 86.5% PPV using the same evaluation set. This is about 10% more accurate than existing phage finding tools. PHAST is freely available at (http://phast.wishartlab.com).

MATERIALS AND METHODS

PHAST is an integrated search and annotation tool that combines genome-scale ORF prediction and translation (via GLIMMER), protein identification (via BLAST matching and annotation by homology), phage sequence identification (via BLAST matching to a phage-specific sequence database), tRNA identification, attachment site recognition and gene clustering density measurements using density-based spatial clustering of applications with noise (DBSCAN) (17) and sequence annotation text mining. In addition to these basic operations, PHAST also evaluates the completeness of the putative prophage, tabulates data on the phage or phage-like features and renders the data into several colorful graphs and charts. Details about the databases, algorithms and implementation are given below.

Creation of custom prophage and bacterial sequence databases

PHAST's prophage sequence database consists of a custom collection of phage and prophage protein sequences from two sources. One is the National Center for Biotechnology Information (NCBI) phage database that includes 46 407 proteins from 598 phage genomes. The other source is from the prophage database (12), which consists of 159 prophage regions and 9061 proteins not found in the NCBI phage database. Since many of the prophage proteins in the prophage database are actually bacterial proteins and some have only been identified computationally, we only selected those prophage proteins that have been associated with a clear phage function. This set includes a total of 379 phage protease, integrase and structural proteins. This PHAST phage library is used to identify putative phage proteins in the query genome via BLASTP (13) searches.

In addition to a custom, self-updating phage sequence library, PHAST also maintains a bacterial sequence library consisting of 1300 non-redundant bacterial genomes/proteomes from all major eubacterial and archaebacterial phyla. This bacterial sequence library contains more than four million annotated or partially annotated protein sequences. Relative to the full GenBank protein sequence library (100+ million sequences), this bacterial-specific library is 25× smaller. This means that PHAST's genome annotation step (see below) can be accomplished 25× faster.

Genome annotation and comparison

PHAST accepts both raw DNA sequence and GenBank annotated genomes. If given a raw genomic sequence (FASTA format), PHAST identifies all ORFs using GLIMMER 3.02 (14). This ORF identification step takes about 45 s for an average bacterial genome of 5.0 Mb. The translated ORFs are then rapidly annotated via BLAST using PHAST's non-redundant bacterial protein library (~2–3 min/genome). Because tRNA and tmRNA sites provide valuable information for identifying the attachment sites, they are calculated using the programs tRNAscan-SE (15) and ARAGORN (16). If an input (GenBank formatted) file is provided with complete protein and tRNA information, these steps are skipped. Phage or phage-like proteins are then identified by performing a BLAST search against PHAST's local phage/prophage sequence database along with specific keywords searches to facilitate further refinement and identification. Matched phage or phage-like sequences with BLAST *e*-values less than 10^{-4} are saved as hits and their positions tracked for subsequent evaluation for local phage density by DBSCAN (17).

Identification of prophage regions and prediction of their completeness

Prophages can be considered as clusters of phage-like genes within a bacterial genome. The primary challenge (after phage-like genes have been identified) is to determine if these genes are sufficiently well clustered or proximal to each other to be considered prophage candidates. Although there are a few reported clustering methods for identifying phage gene clusters (9–11), we found the general DBSCAN algorithm performs just as well, likely because the identification of clusters of prophage genes is not a particularly difficult task. DBSCAN takes two parameters: the cluster size *n* and a distance *e*. The parameter *n* defines the minimal number of phage-like genes required to form a prophage cluster and *e* is the maximal spatial distance between two neighbor genes within the same cluster. In our case, the spatial distance between two genes is just the number of nucleotides between them. In other words, *n* can be considered as the minimal prophage size and *e* is the protein density within the prophage region. Empirically, we set *n* to be 6, since prophages generally have more than five proteins. The value of *e* was set to 3000 based on assessments from a small number of identified prophages in ProphageDB (12). We found that using a moderately different *e*-value will generally not change the prediction sensitivity. If PHAST's input file is an annotated GenBank file, an additional text scan is performed to identify prophages that may not have been found by clustering. This secondary (moving window) scan looks for specific phage-related keywords in the GenBank protein name field of the input file, such as 'protease', 'integrase' and 'tail fiber'. If 6 or more proteins associated with these keywords are found within a window of 60 proteins, the region is considered as a putative prophage region even if an insufficient number of phage-like genes were found by DBSCAN within this region. Finally, if the identified prophage contains an integrase, potential phage attachment sites (one for each integrase in tandem prophages) are then identified by scanning the region for short nucleotide repeats (12–80 bases) (18).

After all prophage regions have been detected, a completeness score is assigned to each identified prophage. Three potential scenarios are considered: (i) the region only contains genes/proteins of a known phage; (ii) >50% of the genes/proteins in the region are related to a known phage and (iii) <50% of the genes/proteins in the region are related to a known phage. In scenario (i), the region automatically has a completeness score of 150 (the maximum). In scenario (ii) and (iii), the region's completeness score is calculated as the sum of the scores corresponding to the region's size and number of genes. If it is found that the region is related to a known phage, both scores are calculated using the size and number of matched genes of the related phage, otherwise they are calculated using the average size (30 kb) and average number of genes (40) of typical phages. The total score in scenario (iii) also counts the number of identified 'cornerstone' genes as well as the density of phage-like genes in the region. 'Cornerstone genes' are genes encoding proteins involved in phage structure, DNA regulation, insertion and lysis (1). Table 1 shows the details of PHAST's completeness score calculation. A prophage region is considered to be incomplete if its completeness score is less than 60, questionable if the score is between 60 and 90, and intact if the score is above 90.

Program and web server characteristics

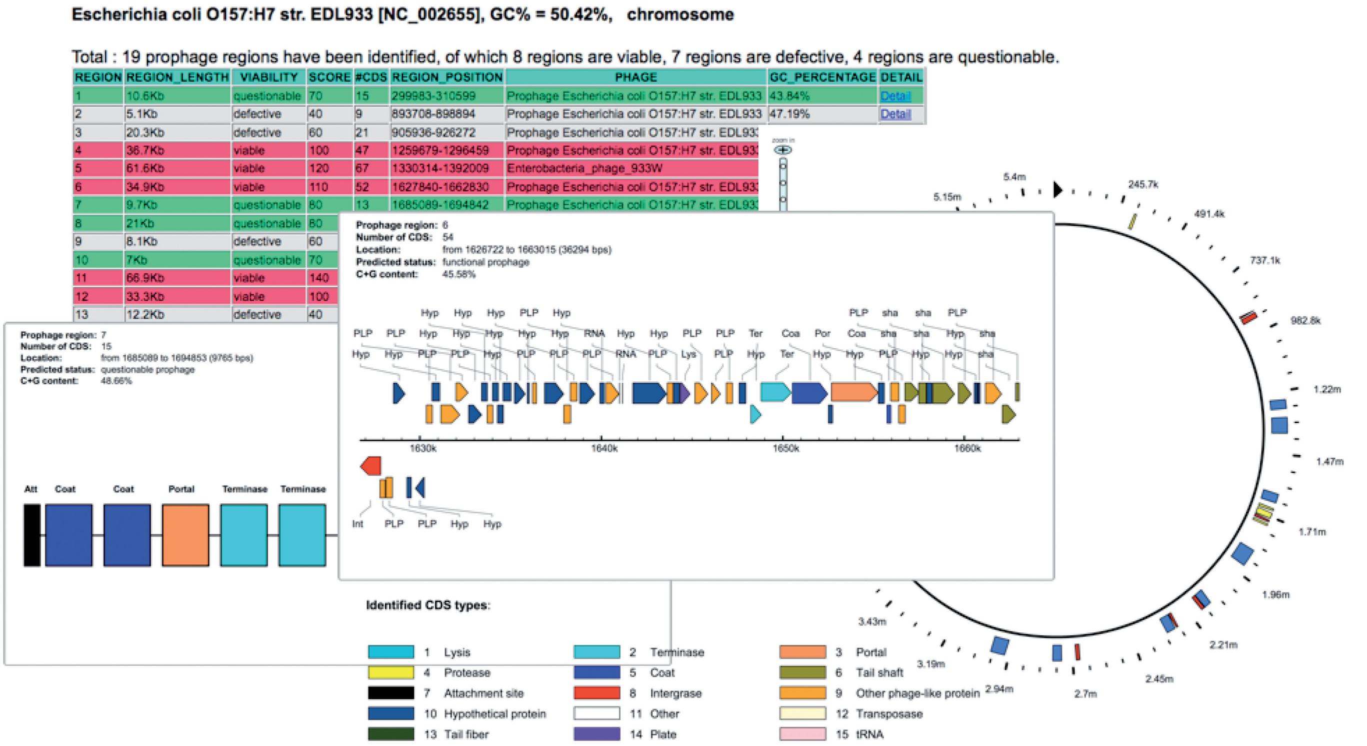
PHAST's search, annotation and DBSCAN clustering software were written using a combination of C and Java. PHAST's web interface was implemented using a standard CGI framework. PHAST's interactive Google-Map style graphics were built using Adobe's Flash Builder. PHAST also supports remote scripting using a URL API (this is described under PHAST's 'Instructions' link) and it maintains a large, hyperlinked database of pre-computed bacterial genomes for rapid prophage identification among known/well-studied genomes (see PHAST's 'Databases' link). A screenshot montage of PHAST's output is given in Figure 1. The web application is platform independent and has been tested successfully on Internet Explorer 8.0, Mozilla Firefox 3.0 and Safari 4.0. However, in order to view the Flash output the user must have Adobe Flash Player installed. This is freely available at http://www.adobe.com/products/flashplayer. For the most up to date instructions of how to use the server please read the online help page at http://phast.wishartlab.com/how_to_use.html.

Table 1. PHAST's phage completeness score calculation

	Scenario A	Scenario B	Scenario C
Number of nucleotides (# bases)	–	(# bases in the region/ # bases in the related phage) × 100	+10 if # bases >30 kb
Number of genes (# genes)	–	(# genes in the region/ # genes in the related phage) × 100	+10 if # genes >40
Cornerstone genes[a]	–	–	+10 for each cornerstone gene
Phage-like genes	–	–	+10 if occupies 70% or more of the region
Final scores	150	Sum of above	Sum of above

The completeness score is calculated for three different scenarios: (A) the region contains all genes of a known phage. (B) >50% of the genes in the region are related to a known phage. (C) <50% of the genes in the region are related to a known phage.

[a]Cornerstone genes are identified key phage structural genes (using keywords such as 'capsid', 'head', 'plate', 'tail', 'coat', 'portal' and 'holin') and phage DNA regulation genes (such as 'integrase', 'transposase' and 'terminase') and phage function genes (such as 'lysin' and 'bacteriocin').



Escherichia coli O157:H7 str. EDL933 [NC_002655], GC% = 50.42%, chromosome

Total : 19 prophage regions have been identified, of which 8 regions are viable, 7 regions are defective, 4 regions are questionable.

REGION	REGION_LENGTH	VIABILITY	SCORE	#CDS	REGION_POSITION	PHAGE	GC_PERCENTAGE	DETAIL
1	10.6Kb	questionable	70	15	299983-310599	Prophage Escherichia coli O157:H7 str. EDL933	43.84%	Detail
2	5.1Kb	defective	40	9	893708-898894	Prophage Escherichia coli O157:H7 str. EDL933	47.19%	Detail
3	20.3Kb	defective	60	21	905936-926272	Prophage Escherichia coli O157:H7 str. EDL93		
4	36.7Kb	viable	100	47	1259679-1296459	Prophage Escherichia coli O157:H7 str. EDL93		
5	61.6Kb	viable	120	67	1330314-1392009	Enterobacteria_phage_933W		
6	34.9Kb	viable	110	52	1627840-1662830	Prophage Escherichia coli O157:H7 str. EDL93		
7	9.7Kb	questionable	80	13	1685089-1694842	Prophage Escherichia coli O157:H7 str. EDL93		
8	21Kb	questionable	80					
9	8.1Kb	defective	60					
10	7Kb	questionable	70					
11	66.9Kb	viable	140					
12	33.3Kb	viable	100					
13	12.2Kb	defective	40					

Prophage region:	6
Number of CDS:	54
Location:	from 1626722 to 1663015 (36294 bps)
Predicted status:	functional prophage
C+G content:	45.58%

Prophage region:	7
Number of CDS:	15
Location:	from 1685089 to 1694853 (9765 bps)
Predicted status:	questionable prophage
C+G content:	48.66%

Figure 1. A screenshot montage of some of PHAST's different graphical and tabular views including its linear and circular genome renderings as well PHAST's corresponding prophage annotation.

Performance evaluation

In order to compare PHAST's performance with other programs, we used a collection of hand-annotated prophages from 54 bacterial genomes (1,10) as our 'gold standard' reference control. PHAST was evaluated using both GenBank annotated sequences (i.e. bacterial genomes with manually or semi-automatically annotated genomes) as well as raw DNA sequence files. The performance was measured using both sensitivity [TP/(TP + FN)] and positive predictive value or PPV [TP/(TP + FP)]. Using this 54 genome data set PHAST achieved, a sensitivity (Sn) of 85.4% and a PPV of 94.2% when evaluated using GenBank annotated files. When using raw DNA sequence and its own ORF finding and genome annotation tools, PHAST achieved a sensitivity of 79.4% and a PPV of 86.5% for the same 54 genomes. PHAST's performance using the same annotated GenBank data was superior to Prophinder (Sn 77.5%, PPV 93.6%), Prophage Finder (Sn 92.1%, PPV 52.1%) and Phage_Finder (Sn 68.5%, PPV 94.3%). PHAST's performance using raw DNA sequence data does not quite match that of the pre-annotated data, but its combined sensitivity/positive predictive value is still comparable to Prophinder and superior to both Prophage Finder and Phage_Finder. Detailed comparisons for both the GenBank and raw sequence inputs for all 54 genomes can be found in PHAST's documentation page (http://phast.wishartlab.com/documentation.html). PHAST's improved performance does not necessarily indicate

Prophinder, Prophage Finder or Phage_Finder's phage finding algorithms are inferior to PHAST's algorithm. Rather, some of the performance gain appears to be due to PHAST's implementation of a newer, larger phage sequence library and perhaps a better exploitation of keyword annotations.

A further challenge with evaluating any kind of prophage identification software is that there is no 'absolute' or 'gold' standard. Careful manual annotation by phage experts is certainly a high standard, but it is more than likely that some prophages in the 54 evaluation genomes were not identified, having decayed or mutated too much for them to appear in the Casjens reference list (1). In other words, some of the false positive predictions may in fact be true positives. Indeed, through manual inspection of PHAST's results we found a number of 'dense' positive BLAST hits to phage proteins in several genomes, but these were not labeled as prophages in the Casjens reference list. Instead of 'false positives', we believe that they should be considered as prophage-related regions that have not been previously reported in the literature.

In addition to evaluating PHAST's prophage identification performance, we also evaluated its speed. Given that PHAST accepts two kinds of file input (raw FASTA DNA sequence and GenBank formatted files), we assessed its performance for both kinds of input files. When given raw genomic sequence, PHAST must run GLIMMER as well as several gene/protein identification programs. Using the raw *Escherichia coli* O157:H7 genome sequence only (GenBank accession NC_002655), PHAST completed its prophage identification in just over 4 min. When tested on the same input file, Prophage Finder returned results after 20 min. However, it is important to note that Prophage Finder does not annotate bacterial genes, its output is very 'crude' and its combined Sn/PPV score is significantly worse than PHAST's (Table 2). Using the GenBank annotated *E. coli* O157:H7 file, PHAST completed its prophage identification in 140 s. Using the same annotated NC_002655 file for the Prophinder (10) web server took 33 min, while using a local copy of Phage_Finder (9) running on a 2.1 GHz Pentium PC with 12 Gb RAM, the same file took 93 min. These data suggest that PHAST is between 5 and 40 times faster than existing prophage finding programs. A more complete feature and performance comparison between PHAST and other existing prophage finding tools is given in Table 2.

Limitations

PHAST is not without some limitations. First, like all other database-driven annotation systems, PHAST obviously performs poorly at identifying novel phages, whose genes/proteins are not closely related to any record in the PHAST database. In this regard, the appearance of large numbers of proximal proteins with unknown function could be a good indication of a novel phage. Second, the DBSCAN algorithm used by PHAST assumes an even density of phage-like hits in every prophage genomic sequence, which is not generally true in practice. Consequently, a highly uneven distribution of phage-like genes could potentially fool the DBSCAN algorithm. Finally, PHAST will occasionally 'split' larger prophages into a number of smaller prophages due to a paucity of BLAST hits.

CONCLUSIONS

PHAST represents a new generation of fast prophage identification and phage annotation tools that produces

Table 2. Features and performance of PHAST relative to other prophage identification tools

	PHAST	PROPHINDER	PROPHAGE FINDER	PHAGE_FINDER
Execution time (*E. coli* O157:H7 5.5 Mb)	140 s	1980 ± 90 s	N/A	5547 s[b]
Execution time (*E. coli* O157:H7 sequence alone)	240 s	N/A	1800 ± 90 s	N/A
Accepts raw DNA file	Yes	No	Yes	No
Accepts Genbank file	Yes	Yes	No	Yes
Sensitivity (%)	85.4	77.5	N/A	68.5[b]
Sensitivity (%) (sequence alone)	79.4	N/A	92.1[a]	N/A
PPV (%)	94.2	93.6	N/A	94.3[b]
PPV (sequence alone) (%)	86.5	N/A	52.1[a]	N/A
Downloadable images	Yes	No	No	No
Phage completeness labeling	Yes	No	No	No
Circular genome view	Yes	No	No	No
Zoomable graphics	Yes	No	No	No
Highlights key phage proteins	Yes	No	No	No
Scriptable operation	Yes	No	No	Yes
Attachment site prediction	Yes	No	No	Yes
Output type	Tables + graphics	Tables + graphics	Text only	Text only
Output readability	Good	Good	Poor	Poor

The performance of all web servers and programs was evaluated using a reference set of 54 manually annotated genomes (1,10). Annotations for all 54 genome inputs can also be found at Prophinder's web page. Phage_Finder was tested locally (strict mode) using HMMER 2.3 for its HMM search. Prophage Finder's (11) web server was tested using its default parameters. Detailed results can be found on the PHAST website.

[a] Prophage Finder failed to return results for all input files. The numbers reported here are for just 46 of the 54 genomes.

[b] Phage_Finder can be run under four different modes, this table reports the results using the strict mode with HMMER 2.3. Using other parameters one obtains: Sn = 88.4%, PPV = 23.7% (HMMER 2.3, non-strict mode), Sn = 90.3%, PPV = 23.9% (HMMER 3.0, non-strict mode) and Sn = 0.0%, PPV = 0.0% (HMMER 3.0, strict mode). When HMMER 3.0 is used Phage_Finder is significantly faster (898 s for *E. coli* O157:H7), but significantly less accruate.

accurate results in minutes using only raw or lightly annotated genome sequence data. PHAST also produces extensive text summaries, downloadable figures, circular and linear genome views as well as colorful, zoomable, user-interactive graphics. As phage and prophage databases continue to expand [with >100 million different phage genomes still to be sequenced (19)], we believe that PHAST's integrated comparative approach to phage finding will only lead to continued improvements in its sensitivity and specificity.

ACKNOWLEDGEMENTS

The authors wish to thank the referees for their helpful suggestions. Y.Z. wrote the phage clustering, the phage identification scripts and the FLASH viewer. Y.L. prepared the phage and bacterial sequence databases, built the web interface and implemented various external programs (BLAST, GLIMMER, CGVIEW). K.H.L. co-designed the methods and tested the server. D.S.W. and J.J.D. conceived the ideas, server requirements and general methodology. All authors contributed to the writing of the manuscript and all have seen and approved of its content.

FUNDING

The authors wish to thank the Canadian Institutes of Health Research (CIHR) and Genome Alberta (a division of Genome Canada) for financial support. Funding for open access charge: Canadian Institutes of Health Research.

Conflict of interest statement. None declared.

REFERENCES

1. Casjens,S. (2003) Prophages and bacterial genomics: what have we learned so far? *Mol. Microbiol.*, **49**, 277–300.
2. Coates,A.R. and Hu,Y. (2007) Novel approaches to developing new antibiotics for bacterial infections. *Br. J. Pharmacol.*, **152**, 1147–1154.
3. Bar,H., Yacoby,I. and Benhar,I. (2008) Killing cancer cells by targeted drug-carrying phage nanomedicines. *BMC Biotechnol.*, **8**, 37.
4. Sullivan,M.B., Waterbury,J.B. and Chisholm,S.W. (2003) Cyanophages infecting the oceanic cyanobacterium Prochlorococcus. *Nature*, **424**, 1047–1051.
5. Fouts,D.E. (2004) Bacteriophage bioinformatics. In Fraser,C.M., Read,T.D. and Nelson,K.E. (eds), *Microbial Genomes*. Humana Press Inc., Totowa, NJ, pp. 71–91.
6. Nicolas,P., Bize,L., Muri,F., Hoebeke,M., Rodolphe,F., Ehrlich,S.D., Prum,B. and Bessières,P. (2002) Mining Bacillus subtilis chromosome heterogeneities using hidden Markov models. *Nucleic Acids Res.*, **30**, 1418–1426.
7. Srividhya,K.V., Alaguraj,V., Poornima,G., Kumar,D., Singh,G.P., Raghavenderan,L., Katta,A.V., Mehta,P. and Krishnaswamy,S. (2007) Identification of prophages in bacterial genomes by dinucleotide relative abundance difference. *PLoS One.*, **2**, e1193.
8. Nelson,K.E., Weinel,C., Paulsen,I.T., Dodson,R.J., Hilbert,H., Martins dos Santos,V.A., Fouts,D.E., Gill,S.R., Pop,M., Holmes,M. *et al.* (2002) Complete genome sequence and comparative analysis of the metabolically versatile Pseudomonas putida KT2440. *Environ. Microbiol.*, **4**, 799–808.
9. Fouts,D.E. (2006) Phage_Finder: automated identification and classification of prophage regions in complete bacterial genome sequences. *Nucleic Acids Res.*, **34**, 5839–5851.
10. Lima-Mendez,G., Van Helden,J., Toussaint,A. and Leplae,R. (2008) Prophinder: a computational tool for prophage prediction in prokaryotic genomes. *Bioinformatics*, **24**, 863–865.
11. Bose,M. and Barber,R.D. (2006) Prophage Finder: a prophage loci prediction tool for prokaryotic genome sequences. *In Silico Biol.*, **6**, 223–227.
12. Srividhya,K.V., Rao,G.V., Raghavenderan,L., Mehta,P., Prilusky,J., Sankarnarayanan,M., Sussman,J.L. and Krishnasswamy,S. (2006) Database and comparative identification of prophages. *Lec. Notes Control Informat. Sci.*, **344**, 863–868.
13. Altschul,S.F., Madden,T.L., Schäffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
14. Salzberg,S.L., Delcher,A.L., Kasif,S. and White,O. (1998) Microbial gene identification using interpolated Markov models. *Nucleic Acids Res.*, **26**, 544–548.
15. Lowe,T.M. and Eddy,S.R. (1997) tRNAscan-SE: a program for improved detection of transfer RNA genes in genomic sequence. *Nucleic Acids Res.*, **25**, 955–964.
16. Laslett,D. and Canback,B. (2004) ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. *Nucleic Acids Res.*, **32**, 11–16.
17. Ester,M., Kriegel,H.P., Sander,J. and Xu,X. (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. *KDD-1996 Proceedings*. AAAI Press, Menlo Park, CA, pp. 226–231.
18. Williams,K.P. (2001) Integration sites for genetic elements in prokaryotic tRNA and tmRNA genes: sublocation prefeence of integrase subfamilies. *Nucleic Acids Res.*, **30**, 866–875.
19. Rohwer,F. (2003) Global phage diversity. *Cell*, **113**, 141.