Accelerating Evolutionary Molecular Phylogenetic analyses on the NUS TCG Grid

Hu Yongli
Graduate Research Student, Dept of Biochemistry, Yong Loo Lin School of Medicine

Introduction

Phylogenetics is the science of estimating the evolutionary past. In the case of molecular phylogeny, evolutionary events and epidemiology can be studied based on the comparison of DNA or protein sequences (1). The most commonly used package of programs for inferring phylogenies is PHYLIP (the PHYLogeny Inference Package). Developed during 1980s, PHYLIP is one of the most widely-distributed phylogeny packages used for building the largest number of published phylogenetic trees and currently, has with more than 15,000 registered users (2) worldwide. The PHYLIP package is freely available over the Internet, and has been written to work on many different kinds of operating systems such as Windows, Mac and Linux. Methods in the package include parsimony, distance matrix, and likelihood methods. Particularly important is the bootstrapping and consensus trees programs which are vital tools for scientists to build phylogentic trees for the inference of molecular evolutionary events. In addition, PHYLIP is also able to handle different data types which include Deoxyribonucleic Acid (DNA) and protein sequences, gene frequencies, restriction sites and fragments, distance matrices, and discrete characters (3) to suit different users with their specific needs for phylogentic analyses.

In a typical protein phylogentic tree building process (see figure 1), the protein sequence alignment file (presented in .phy format) is placed into the program *seqboot* in PHYLIP, where the program reads in the input data set, and produces multiple data sets (typically 100) by bootstrap resampling of the input data. Next, the output from *seqboot* is subsequently used by the program *protdist*, where distance measures for protein sequences are calculated and are required for downstream tree building. Outputs generated from *protdist* will be subsequently used by the program *neighbor*, where the clustering of sequences within each individual dataset can be carried out before merging the *neighbor* datasets for the final program *consense* where the a consensus tree can be built with the 100 datasets. Finally, the visualization of the *consense* output can be carried out with *drawgram* (a program in PHYLIP) or any phylogenetic tree visualization program like treeview (4).

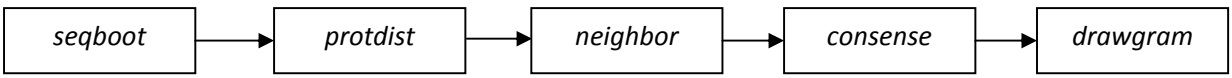


Figure 1. Typical pipeline for building a protein phylogentic tree with PHYLIP. The output of one program is used as the input for the next program. For example, the output for *seqboot* is used as the input for the program *protdist* and the output from the program protdist is used by *neighbor*.

For phylogenetic analyses that deal with relatively small protein sequence datasets (e.g. datasets with less than 50 sequences), the time required for running the whole PHYLIP pipeline will be extremely fast. However, as the number of sequences increases, the runtime will increase significantly. A preliminary study on a dataset of 178 protein sequences running a typical tree-building process, as in Figure 1, on a Sunfire 6800 server, with 16 CPUs at 900MHz and 16GB RAM, requires more than 9 hours to finish the tree-building process. In particular, the *protdist* program took the longest running time of 9 hours and 6 minutes to complete while other programs like *seqboot*, *neighbor* and *consense* each only takes less than 2 minutes to complete. Having

identified the main rate determining step in protein phylogeny using PHYLIP, the next immediate question to answer will be that if data processing with *protdist* can be accelerated. The answer will be yes and there are two major options to be considered. The first option is to increase the efficiency of the computers/servers used for analyses like getting a server with higher processor speed and a larger RAM. However, this is not the best option as the purchase of these high end machines is very expensive and this is not cost effective for many research groups that rely on limited fundings. Then, the next option will be to parallelize the *protdist* process and place the parallelized program on readily available grid architecture for processing. This approach fits perfectly in the picture as, first, each of the 100 *seqboot* output datasets can be discretely used for the calculation of protein distances in *protdist* and this allows the 100 datasets within the *seqboot* outputs to be split and disseminated to multiple grid client machines for processing before merging the 100 different datasets for analysis by *neighbor*. Second, *protdist* serialization can be carried out with the availability of readily established grid architecture in NUS.

Thus, the report will document the process of placing *protdist* on the NUS TCG grid as an application called meta-PHYLIP and evaluating the speedup achieved for molecular phylogenetic analyses of large number of protein sequences by meta-PHYLIP.

Materials and Methods

Preparing the *protdist* program in meta-PHYLIP

The PHYLIP suite 3.68 was downloaded from http://evolution.genetics.washington.edu/phylip/. The linux source code was first complied on a Linux-based server, with Intel Pentium 4 CPU 3.00GHz, 4 GB of RAM running on Slackware 10.0, to produce the executable version of *protdist*, the main program running on meta-PHYLIP. After the successful testing of meta-PHYLIP, together with the parameter files, on the NUS altas-4 linux computer cluster, the executable *protdist* is ready to be placed on the NUS TCG Grid with the application name meta-PHYLIP.

Preparation of .PHY Datasets for input into *seqboot* program

Two different types of datasets were prepared for analysis.

The first group of data includes seven input datasets, ranging from 100 to 5000 sequences, created from an alignment of Human Immunodeficiency Virus Type 1(HIV-1) Clade D *vif* protein. These seven input files were used to test effect of the different sequence numbers, while keeping the protein alignment length the same, on the speed-up of meta-PHYLIP on the NUS TCG Grid.

Next, data processing of real biological protein sequence alignments from HIV-1, Dengue virus and Influenza A virus were also carried out with meta-PHYLIP. In this second group of data, the speedup of biological research data processing with meta-PHYLIP was explored.

Preparing *seqboot* outputs and parameter files for meta-PHYLIP

All the protein alignments (in .phy format) used for analysis were first sent to the *seqboot* program to produce 100 data sets by bootstrapping. A simple shell script, chop_seqboot.sh (refer to Insert 1 in Appendix), was subsequently used to split the 100 datasets into 100 different files, each having the filename as input1.dat.00000X (where X ranges from 1-100, each file contains one of the 100 datasets produced by

seqboot). In addition, 100 parameter files (named as input2.dat.00000X, where X ranges from 1-100) were created. These parameter files serve as instructions files essential for the *protdist* program to process each of the input1.dat.00000X files. Leveraging on the existing parametric study program available on the NUS TCG Grid, 100 identical parameter files will be required for meta-PHYLIP to function.

Running meta-PHYLIP

The 200 files, 100 datasets from splitting *seqboot* output and 100 instructional parameter files for *protdist*, were zipped before sending to the NUS TCG Grid for processing. On the Grid server, the zipped file will be unzipped and repackaged to 100 new packages, each package will consist of a *seqboot* output file input1.dat.000001 (for example), a parameter file input2.dat.000001 and a *protdist* program. Subsequently, these new packages will be dispatched to the next available grid client from processing. To submit the files to the grid, the parametrics study program needs to be downloaded from http://puma1.nus.edu.sg/svu/tcg/apps/. The program name on the NUS TCG Grid is called meta-PHYLIP. An example for the command required to run the HIV-1 Clade B *gag* protein is as shown below in figure 2.

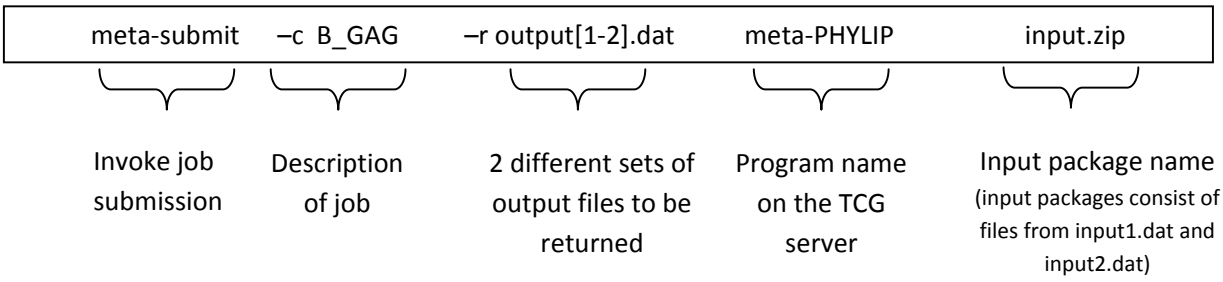


Figure 2. The sample command required for data-processing for meta-PHYLIP on the NUS TCG Grid. Output1 returned will be the 100 result sets (output1.dat.000001-100) from *protdist* while output2.dat will contain the 100 log files generated in the format output2.dat.000001-100.

Upon completion of data processing with meta-PHYLIP, two different sets of outputs will be returned to the user. The first set of outputs, in the format, output1.dat.00000X (where X ranges from 1 to 100), are the outputs from *protdist* and these outputs can be merged and subsequently used for processing by program *neighbor* (refer to pipeline in figure 1). The second set of outputs, in the format output2.dat.00000X (where X ranges from 1 to 100), are the log files created and these are essential tools for the user to determine whether the *protdist* process have been running smoothly with the parameter files they have submitted.

Evaluating the speedup of meta-PHYLIP

Total running time (RT_{100}) and total CPU time (Tp) were recorded (in seconds) where RT_{100} is the time from the job creation to return of the last output to the grid server and Tp is the total CPU time required to run the program in serial. To quantify the acceleration of data processing on the grid, the following equation (1) is used to obtain the speedup

$$\text{Speedup} = RT_{100} / Tp \qquad \text{Equation (1)}$$

Results and Discussions

Grid-enabling of PHYLIP

Leveraging on the advantage to individually run each dataset, out of 100, of the *seqboot* output with *protdist*, the grid-enabled *protdist* (meta-PHYLIP) will take the 100 pre-split *seqboot* outputs and distribute them to 100 grid clients in the NUS TCG Grid for processing. The steps undertaken to enable protein pyhlogenetic analysis by meta-PHYLIP is as shown in figure 2. First, the user will generate a *seqboot* output file, of 100 protein sequence datasets, from a protein sequence alignment. Next, the *seqboot* output file will be divided individually into 100 *seqboot* resampled protein sequence alignment files using the program chop_seqboot.sh (Refer to Insert 1 in Appendix) and each file will contain one output from *seqboot*. Then, each dataset, containing one protein sequence dataset from chop_seqboot.sh and a parameter file, will be uploaded to GridMP server (koala1), running the Univa UD Grid MP platform middleware, which will dispatch the input data file, together with the parameter files and the *protdist* program, to the available grid client machines for processing. Upon completion, the *protditst* outputs will be sent back to koala1 for the user to download.

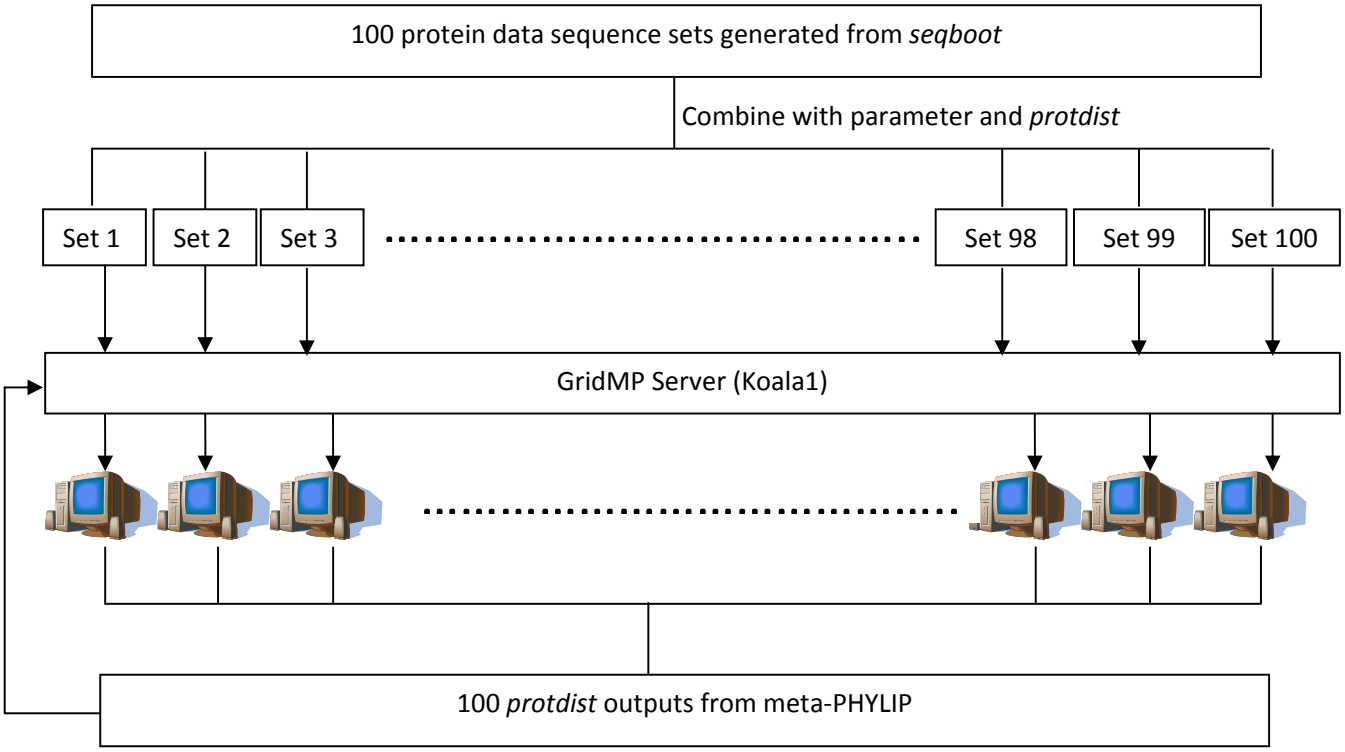


Figure 3. A simplified schematic representation of the flow of data for grid-enabling of data processing by software *seqboot*.

The meta-PHYLIP application has two interesting features. First, the application allows the user to customize their parameters file in order to achieve the desired data processing by *protdist*. However, the customization of the parameter file requires the user to have a fundamental understanding of how the *protdist* program functions. An example of the parameter file is shown in figure 4 and how the information of the parameter file is fed into the *protdist* program is shown in figure 4(ii). The list of available parameter fields to be manipulated are shown in figure 5.

(i)

Parameter File

input1.dat
F
output1.dat
Y

(ii)

```
protdist.exe: can't find input file "infile"
Please enter a new file name> input1.dat

protdist.exe: the file "outfile" that you wanted to
     use as output file already exists.
     Do you want to Replace it, Append to it,
     write to a new File, or Quit?
     (please type R, A, F, or Q)

F
Please enter a new file name> output1.dat

Protein distance algorithm, version 3.69

Settings for this run:
  P  Use JTT, PMB, PAM, Kimura, categories model?  Jones-Taylor-Thornton matrix
  G  Gamma distribution of rates among positions?  No
  C            One category of substitution rates?  Yes
  W                     Use weights for positions?  No
  M                   Analyze multiple data sets?  No
  I                  Input sequences interleaved?  Yes
  0                Terminal type (IBM PC, ANSI)?  IBM PC
  1             Print out the data at start of run  No
  2           Print indications of progress of run  Yes

Are these settings correct? (type Y or the letter for one to change)
Y
```

Figure 4. (i) The parameter file required for sending instructions to the *protdist* program to run with the default parameters with the input file input1.dat (which is one of the *seqboot* output files) and the output will be written to output1.dat. (ii) Demonstration of how the parameters will be entered into a *protdist* program while running *protdist* on a computer with Windows XP operating system. The parameters in 4(i) are boxed up in red.

```
Settings for this run:
  P  Use JTT, PMB, PAM, Kimura, categories model?  Jones-Taylor-Thornton matrix
  G  Gamma distribution of rates among positions?  No
  C            One category of substitution rates?  Yes
  W                     Use weights for positions?  No
  M                   Analyze multiple data sets?  No
  I                  Input sequences interleaved?  Yes
  0                Terminal type (IBM PC, ANSI)?  IBM PC
  1             Print out the data at start of run  No
  2           Print indications of progress of run  Yes

Are these settings correct? (type Y or the letter for one to change)
```

Figure 5. Parameters available for manipulation in the *protdist* program. The *protdist* parameter file requires the user to have knowledge on the manipulation of the program (For an example of the parameter file please see figure 4(i)).

The next interesting feature is that meta-PHYLIP also returns the corresponding log files (see supplementary figure 1) for the *protdist* processing. These files are essential as they help the user verify whether the parameter files submitted are correct for the desired *protdist* analysis and if not, the log files aid in the debugging process to modify the parameters for the next *protdist* execution.

Dataset used for running meta-PHYLIP

Two different groups of protein sequences were used for analysis. The first group of sequences (refer to table 1) consisted of pseudo-datasets made from HIV-1 Clade D *vif* protein alignments. These datasets are used to determine the speedup achieved for different number of protein sequences with the same protein alignment

length. Next, the speedup is analyzed with datasets of real-life biological research data. The name and number of sequences in the datasets are as shown in table 2.

Table 1. Name of dataset and the corresponding number of sequences of the datasets used to analyze the impact of speedup on different number of protein sequences with same protein alignment length.

Name of Dataset	Number of sequences
HIV-1 Clade D *vif_100*	100
HIV-1 Clade D *vif_500*	500
HIV-1 Clade D *vif_1000*	1000
HIV-1 Clade D *vif_2000*	2000
HIV-1 Clade D *vif_3000*	3000
HIV-1 Clade D *vif_4000*	4000
HIV-1 Clade D *vif_5000*	5000

Table 2. Name of dataset and the corresponding number of sequences of the real-life biological datasets used for analysis.

Name of Dataset	Number of sequences
HIV-1 Clade D *vif* protein	178
HIV-1 Clade D *vpr* protein	178
HIV-1 Clade D *gag* protein	1001
HIV-1 Clade D *pol* protein	1660
DENV *Envelope* protein	3845
HIV-1 Clade B gag protein	6412
Influenza A *Hemagglutinin* protein	7001

Studying the effect of speedup using meta-PHYLIP

After establishing meta-PHYLIP on the NUS TCG Grid, the next important step is to determine the speedup capacity of meta-PHYLIP. In total, two different sets of data are used for the analysis. The first group of data consists of a set of seven pseudo alignments generated from HIV-1 Clade D v*if* Protein while the next group contains a seven sets of real-life research biological protein sequence alignments of HIV-1, Dengue Virus and Influenza A virus.

Speedup achieved with protein alignments of different sequence number

The speedup achieved ranges from 14.1 to 65.0 times as compared to running the different Clade D *vif* datasets on standalone machines (See figure 6). For more information on the grid running time, total CPU time and speedup of different number of protein sequences, please refer to Supplementary table 1. In general, the speedup achieved is highly favorable. For example, a dataset of 5000 protein sequences will take more than 3 months to finish running *protdist* on standalone computers but when the same dataset is ran on the grid, it will only take about 2 days, having accelerated the analysis process by more than 60 times. In addition, it is interesting to note that the speedup for small datasets, e.g. 100 protein sequences is relatively lower than that of the larger datasets. Also, caution is required in the interpretation of the results as applications running on the NUS TCG Grid harvest on idle compute cycles of grid clients and the speedup of program execution relies heavily on the availability of idle compute cycles for computation.

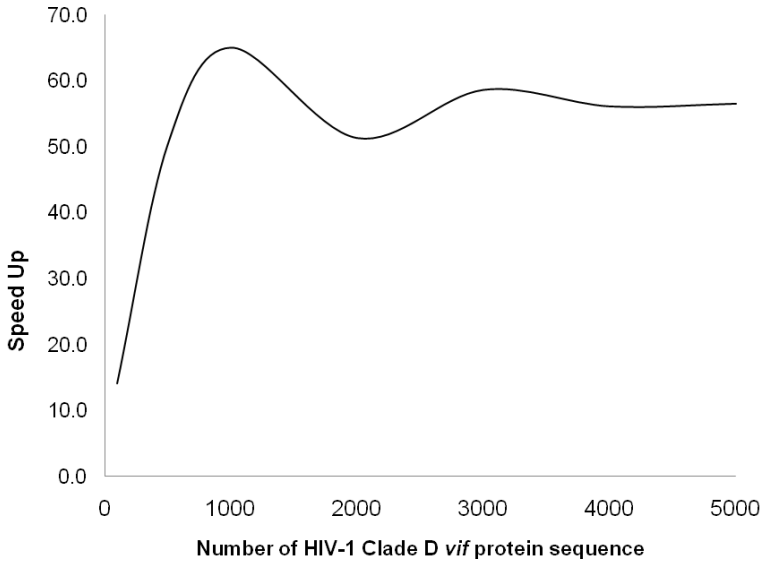


Figure 6. Speedup for running pseudo-datasets of Clade D HIV-1 *vif* protein sequence alignments with meta-PHYLIP.

Speedup achieved for real-life biological research data

On average, the speedup of the real life research data is 45 times as compared to running the different datasets on standalone machines (See figure 7). For more information on the grid running time, total CPU time and speedup of different number of protein sequences, please refer to Supplementary table 2. Also observed is that speedup for small datasets, e.g. 100 protein sequences is relatively lower than that of the larger datasets, which, again, might be related to the availability of idle computer cycles on grid clients.

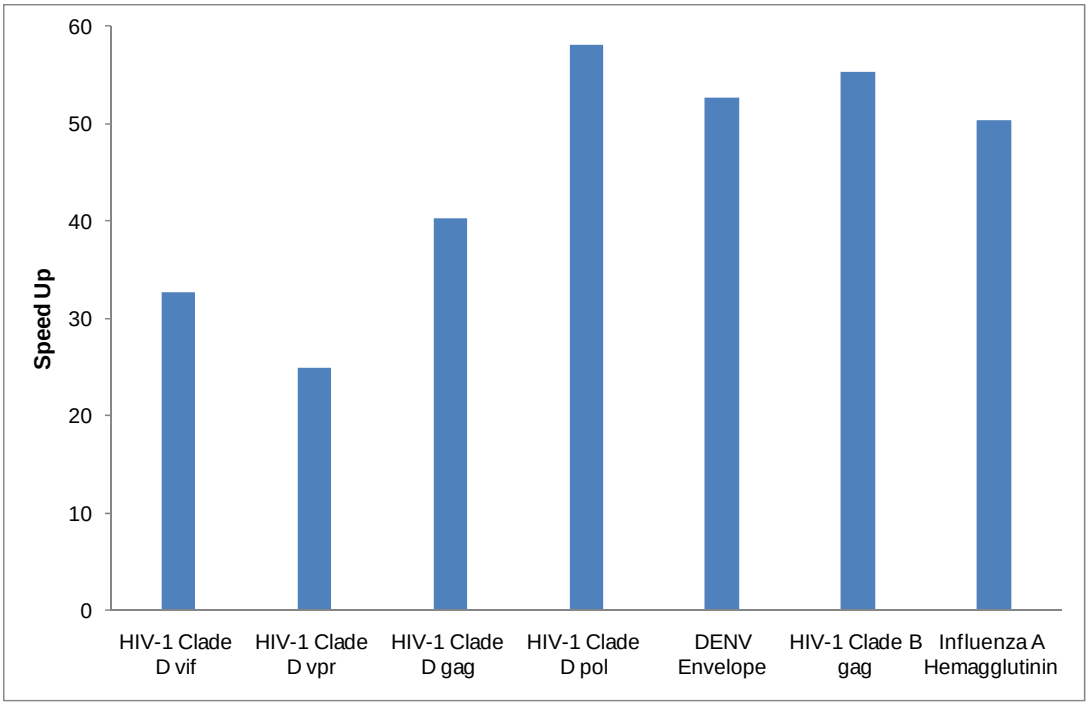


Figure 7. Speedup for running real-life biological research datasets with meta-PHYLIP.

Contributions of meta-PHYLIP on biological research

Molecular phylogeny has extremely useful applications ranging from the study of epidemiology of acquired immune deficiency syndrome to uncovering the origin of life (5). However, with the advancement of sequencing techniques, the amount of sequence data available for phylogenetic analyses has increased exponentially such that phylogenetic analyses can no longer be carried out within an acceptable time frame (6). Molecular phylogeny analysis acceleration by meta-PHYLIP will not only renders extremely large sequence datasets, which previously have been left unanalyzed, analyzable, but it will also allows rapid construction of protein phylogenetic trees which are important in molecular phylogenetic studies of disease outbreaks and emerging pandemics where the identification of viral ancestry and source have important applications in disease treatment and, also, pandemic containment.

Conclusion

A version of PHYLIP has been successfully placed on the NUS TCG Grid as an application called meta-PHYLIP. Molecular phylogenetic analyses with meta-PHYLIP using 14 different datasets had yielded significant speedup of about 14 to 65 times. The acceleration in building phylogenetic trees is extremely important as it enhances the current rate of molecular phylogenetic analyses in biological research settings and, also, allows the analysis of large datasets which have previously been left unanalyzed. Molecular phylogenetic analyses can be completed within a much shorter time frame and alleviates the major bottle-neck in disease epidemiology and evolutionary studies with large number of protein sequences in this era of sequence data explosion.

Acknowledgements

I will like to thank A/P Tan Tin Wee for conceiving the idea of parallelizing PHYLIP on the grid. Also, I will like to express my gratitude to Mr Mark De Silva and Mr Lim Kuan Siong for providing technical support and Mr Mohammad Asif Khan and Miss Heiny Tan for providing their Dengue virus and Influenza A protein sequence alignments. Last but not least, I will like to thank Mr Wang Junhong for his help rendered for placing meta-PHYLIP on the NUS TCG grid.

References

(1) Baldauf, S.L. Phylogeny for the faint of heart: a tutorial. (2003) *Trends Genet.* **16**(6):345-51
(2) Abdennadher, N. and Boesch, R. Porting PHYLIP phylogenetic package on the Desktop GRID platform XtremWeb-CH. (2007) *Stud Health Technol Inform*. **126**:55-64.
(3) General Description of PHYLIP. http://evolution.genetics.washington.edu/phylip/general.html , retrieved on 14 September 2009
(4) Page, R. D. M. TREEVIEW: An application to display phylogenetic trees on personal computers. (1996) *Com Appl in the Biosci.* **12**: 357-358.
(5) Hillis, D.M., Huelsenbeck, J.P. and Cunningham, C.W. Application and accuracy of molecular phylogenies. (1994) *Science.* **264**(5159):671-677
(6) Sanderson, M.J. and Driskell, A.C. The challenge of constructing large phylogenetic trees. (2003) *Trends Plant Sci.* **8**(8):374-379

APPENDIX

Insert 1. Shell script chop_seqboot.sh.

```
#!/bin/sh

num=100
infile=$1 #Name of seqbooot output file#
if [ "$infile" = "" ]; then
  echo "Use command as:"
  echo " > $0  seqboot_output_file"
  echo
  exit 1
fi

TMP=${infile}_$$
/bin/cp -f $infile $TMP
echo "Temp seqboot output file: $TMP"

N=` wc -l $TMP|awk '{print $1}'`
lines=` expr $N / $num `

for i in `seq -w 1 100`; do
  echo "creating file input1.dat.000$i"

  nh=`expr $i \* $lines`
  head -n $nh $TMP|tail -n $lines > input1.dat.000$i
done

/bin/rm -f $TMP
echo "done successfully."
```

```
------------------------
Program Start at
2009091812
------------------------
================================
|  PHYLIP on NUS Grid          |
|  for [protdist + neighbor]   |
================================
Fri Sep 18 12:49:22 SGT 2009
--------------------------------
/usr/local/UD/mpagent/program-6BF0712C-A40E-11DE-883B-0013724CF8BF/pm

untar the phylip.tar exe package now...

Fri Sep 18 12:49:22 SGT 2009
protdist starting ...
--------------------------------------------------
./protdist < input2.dat
←[2J-[Hprotdist:  can't find input file "infile"
Please enter a new file name>
protdist: the file "outfile" that you wanted to
     use as output file already exists.
     Do you want to Replace it, Append to it,
     write to a new File, or Quit?
     (please type R, A, F, or Q)
Please enter a new file name>
←[2J-[H
Protein distance algorithm, version 3.68

Settings for this run:
  P  Use JTT, PMB, PAM, Kimura, categories model?  Jones-Taylor-Thornton matrix
  G  Gamma distribution of rates among positions?  No
  C            One category of substitution rates?  Yes
  W                     Use weights for positions?  No
  M                  Analyze multiple data sets?  No
  I                 Input sequences interleaved?  Yes
  0                Terminal type (IBM PC, ANSI)?  ANSI
  1            Print out the data at start of run  No
  2          Print indications of progress of run  Yes

Are these settings correct? (type Y or the letter for one to change)

Computing distances:
  AAA45059.1
  AAZ06086.1
  BAA20316.1
  ABW72783.1   ..........
  AAX86751.1   ..........
  AAC02418.1   ..........
  CAA83106.1   ..........

Output written to file "output1.dat"

Done.
```

Supplementary Figure 1. An excerpt of the sample output of the log file generated when the program *protdist* is successfully executed in the meta-PHYLIP application. It should be noted that if the output file is successfully generated, the last output will show "Output written to file "output1.dat" Done." (see area boxed up in red)

Supplementary table 1. Raw table of results showing the start, end time, meta-PHYLIP running time, CPU time and Speed-up rate for different number of protein sequences with same protein alignment length.

Name of dataset	Number of sequences	Start Time [a]	End Time [a]	Running Time (Seconds)	Total CPU Time (Seconds)	Speed-up Rate [b]
HIV-1 Clade D *vif_100*	100	2009-09-18 04:47:07	2009-09-18 04:51:25	259	3662	14.1
HIV-1 Clade D *vif_500*	500	2009-09-18 10:19:16	2009-09-18 10:50:05	676	93066	50.3
HIV-1 Clade D *vif_1000*	1000	2009-09-20 11:24:35	2009-09-20 13:15:29	5754	374295	65.0
HIV-1 Clade D *vif_2000*	2000	2009-09-20 11:31:36	2009-09-20 19:31:49	28813	1478121	51.3
HIV-1 Clade D *vif_3000*	3000	2009-09-20 21:45:47	2009-09-21 13:36:16	57029	3344560	58.6
HIV-1 Clade D *vif_4000*	4000	2009-09-21 16:45:19	2009-09-22 22:00:37	105282	5904548	56.1
HIV-1 Clade D *vif_5000*	5000	2009-09-23 06:33:09	2009-09-25 03:23:28	161419	9120945	56.5

[a] Start and end time in UTC.

[b] Please refer to equation (1) for the calculation of Speed-up rate.

Supplementary table 2. Raw table of results showing the start, end time, meta-PHYLIP running time, CPU time and Speed-up rate for datasets of real-biological research.

Name of dataset	Number of sequences	Start Time [a]	End Time [a]	Running Time (Seconds)	Total CPU Time (Seconds)	Speed-up Rate [b]
HIV-1 Clade D *vif*	178	2009-08-12 16:37:06	2009-08-12 16:43:04	358	11720	32.7
HIV-1 Clade D *vpr*	178	2009-08-12 16:43:05	2009-08-12 17:30:14	467	11664	25.0
HIV-1 Clade D *gag*	1001	2009-08-12 23:20:21	2009-08-13 00:10:10	2989	120514	40.3
HIV-1 Clade D *pol*	1660	2009-08-12 17:45:16	2009-08-12 20:00:05	8089	470337	58.1
DENV *Envelope*	3845	2009-08-27 07:37:13	2009-08-28 22:10:24	138791	7299989	52.6
HIV-1 Clade B *gag*	6412	2009-08-25 13:38:08	2009-08-26 13:16:15	84113	4651568	55.3
Influenza A *Hemagglutinin*	7001	2009-09-03 09:57:53	2009-09-11 02:54:13	663500	33394702	50.3

[a] Start and end time in UTC.

[b] Please refer to equation (1) for the calculation of Speed-up rate.