# Integrated Automatic Workflow for Phylogenetic Tree Analysis Using Public Access and Local Web Services

**Kasikrit Damkliang[1,*], Pichaya Tandayya[2], Unitsa Sangket[3] and Ekawat Pasomsub[4]**

[1] Innovative Information Technology for Health Science and Society Research Unit (INTACT) and Department of Information and Communication Technology, Faculty of Science, Prince of Songkla University, Hat Yai, Songkhla, 90112 Thailand

[2] Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University, Hat Yai, Songkhla, 90112 Thailand

[3] Center for Genomics and Bioinformatics Research, Faculty of Science, Prince of Songkla University, Hat Yai, Songkhla, 90112 Thailand

[4] Department of Pathology, Faculty of Medicine, Ramathibodi Hospital, Mahidol University, Bangkok, 10400 Thailand

## Summary

At the present, coding sequence (CDS) has been discovered and larger CDS is being revealed frequently. Approaches and related tools have also been developed and upgraded concurrently, especially for phylogenetic tree analysis. This paper proposes an integrated automatic Taverna workflow for the phylogenetic tree inferring analysis using public access web services at European Bioinformatics Institute (EMBL-EBI) and Swiss Institute of Bioinformatics (SIB), and our own deployed local web services. The workflow input is a set of CDS in the Fasta format. The workflow supports 1,000 to 20,000 numbers in bootstrapping replication. The workflow performs the tree inferring such as Parsimony (PARS), Distance Matrix - Neighbor Joining (DIST-NJ), and Maximum Likelihood (ML) algorithms of EMBOSS PHYLIPNEW package based on our proposed Multiple Sequence Alignment (MSA) similarity score. The local web services are implemented and deployed into two types using the Soaplab2 and Apache Axis2 deployment. There are SOAP and Java Web Service (JWS) providing WSDL endpoints to Taverna Workbench, a workflow manager. The workflow has been validated, the performance has been measured, and its results have been verified. Our workflow's execution time is less than ten minutes for inferring a tree with 10,000 replicates of the bootstrapping numbers. This paper proposes a new integrated automatic workflow which will be beneficial to the bioinformaticians with an intermediate level of knowledge and experiences. The all local services have been deployed at our portal `http://bioservices.sci.psu.ac.th`.

[*] To whom correspondence should be addressed. Email: kasikrit.d@psu.ac.th

doi:10.2390/biecoll-jib-2016-287

# 1 Introduction

In works concerning bioinformatics, although there are many tools, websites, and web services, works in this field still need to be done manually as there still are many gaps and not enough interfaces due to many formats and implementations involved. Cutting and pasting between websites, tools, and services usually induce errors and mistakes including works in phylogenetic tree analysis. There are many conventional software packages and web services utilizing for the phylogenetic trees analysis such as PHYLIPNEW applcations [1][2], MEGA [3], European Bioinformatics Institute (EMBL-EBI) services [4], Swiss Institute of Bioinformatics (SIB) services [5], and cooperation related-tools, for example, an R interface for PHYLIP [6], analytic tools for phylogenetic tree based on powered grid resources [7] and the integrated platform of response web tools for expert and non-expert users [8][9].

## 1.1 Taverna

State-of-the-art scientific methods have been introducing automatic-oriented execution frameworks on various platforms for a decade. For example, the myGrid project proposes Taverna [10][11], a scientific workflow management system (SWFMS) for composing automatic workflows emerged in bioinformatics. Nowadays, Taverna workbench has been widely deployed in a variety of research fields including biodiversity [12], chemistry, astronomy [13], data and text mining, digitization, document and image analysis, etc. Many Taverna distributions are open source and support for a variety of running environments including the Workbench, desktop client application, the Command Line Tool for a quick execution of workflows from a terminal, the Server for remote execution of workflows, the Player (a web interface plugin for submitting remote execution of workflows), and Taverna Online providing researchers to create Taverna workflows from a web browser.

A workflow is a representative of instruction steps which execute and produce required results using various types of services including WSDL Web Services, local scripts, BioMart data warehouses, RESTful Web Services, Grid Services, Cloud Services, R-scripts and distributed command-line scripts [14]. In this paper, we demonstrate our *in-silico* methods using web services and workflow mechanisms. The workflows are composed, saved as workflow scripts (.t2flow) and run on the Taverva workbench. Components of the workflows are web services which are both distributed public-access and our own local services. There also are other SWFMSes in other significant fields such as Kepler for physics [15], Swift for climate science, Vistrails is for earth science, and VIEW for medical science [16]. In addition, the contemporary trend usually is to merge SWFMSes into Cloud platforms and enable users to access services via their portals [17][18].

## 1.2 Problems and Motivations

In our previous work [19], we have proposed an automatic Taverna workflow for shrimp's Single Nucleotide Polymorphism (SNP) analysis. Continuously, large coding sequence (CDS) has been discovered. Software, analysis processes, and related tools have also been developed and upgraded simultaneously. Some kind of software and services employed in the past cannot handle the ever gaining data and processes. For example, it requires at least a thousand of replicates in bootstrapping steps for practical analysis with the CDS and the problem is the dataflow size limitation in the datalink of the Taverna workflow. We have found that it is possible for a Taverna workflow to handle a larger dataflow size according to our local-setting experiments. However, Taverna Workbench cannot support the extensive analysis with its ordinary specification for remote services in the network.

In addition, there is another problem occurring at the server side. For example, a complicated workflow which is data intensive may generate large intermediate results being left on the services server. The server then faces a lack of device space even though the user has sent a command to inform it to release outputs which are no longer needed. As a result, the user has to wait until the server's storage has been actually released and become available once again for the services.

Furthermore, there is a portal that supports a large number of bootstrapping replicates [7] but the bioinformaticians have to prepare their dataset according to the portal supported format and perform dataset preprocessing by themselves before they submit the preprocessed dataset to the portal. These processes are time consuming and tend to cause errors.

## 1.3 Our Proposed Approach

In this paper, we propose new practical steps and automatic Taverna workflows for the phylogenetic tree analysis using various tools and services. Our work includes public access of update and high performance web services at the EMBL-EBI and the SIB, local web services wrapping the PHYLIPNEW applications, and helper services bundled from the Taverna Workbench. We have implemented, built local services, tested and run experiments using Dengue Viruses CDS as the case study dataset. Our proposal supports CDS of both nucleotide and amino acid sequences.

## 1.4 Case Study Dataset

We use the CDS, fetched from the NCBI RefSeq nucleotide database [20]. Accession numbers GenBank: KF744397 - KF744408 are found in Philippines for a 12-control-group and GenBank: JN697058 is found in Malaysia for an out-group. Therefore, our case study contains a 13-CDS dataset and each CDS contains about 10,000 nucleotide-bases.

# 2 Related works

In bioinformatics, phylogenetic tree analysis is usually decomposed into three main steps consisting of multiple sequence alignment (MSA), tree inferring, and tree visualization. In this section, we describe our new practical approach in the phylogenetic tree analysis workflow shown in Figure 1.

## 2.1 Preprocessing Data

The MSA is the preprocessing data for bootstrapping phylogenetic analysis. Both nucleotide and amino acid sequences are supported. There are many MSA tools publicly provided such as the EMBL-EBI, *ClustalOmega* [21], *Kalign* [22], *MAFFT (Multiple Alignment using Fast Fourier Transform)* [23], and *MUSCLE* [24]. In this work, we implicitly utilize the SOAP *ClustalOmega* service for the MSA process. In case of amino acid alignment, our proposed workflow provides sequence translation using the SIB web service. In addition, for general analysis of tree inferring steps, bioinformaticians usually remove some gaps amongst sequences of the MSA result for more tree inferring accuracy, before inputting them into the bootstrapping step. This is an important step because the MSA result significantly influences the outputs of the phylogenetic tree. However, currently, there is no program or service to do the *in-silico* step automatically. The bioinformaticians have to manually determine and trim the MSA result. For the proposed workflow, we automate this process by removing prefix and suffix gaps of the MSA result using our trimming algorithms, and further deploy them as web services.

## 2.2 Tree Inferring Algorithms

The tree inferring process is a significant key for estimating the phylogenetic tree. Its input is the MSA result in the Phylip format from the *ClustalOmega* web service. For the proposed workflow, our bioinformaticians introduce a new adaptive and configurable method for algorithms selection based on the MSA similarity score. We estimate the score using our proposed formula according to Equation 1, where $s$ is the similarity score, $m$ is the number of CDS set, $n$ is the number of members in each CDS set, and $x_{ij}$ is each Percent Identity Matrix (PIM) [21][25] value. We are interested in popular tree inferring algorithms bundled with the PHYLIPNEW [1] applications, EMBOSS [2] conversions of the program in Joe Felsenstein's PHYLIP package which consists of Parsimony (PARS), Distance Matrix - Neighbor Joining (DIST-NJ), and Maximum Likelihood (ML). These inferring algorithms support both nucleotide and amino acid sequences.

$$\text{Equation 1: } s = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}}{mn}$$

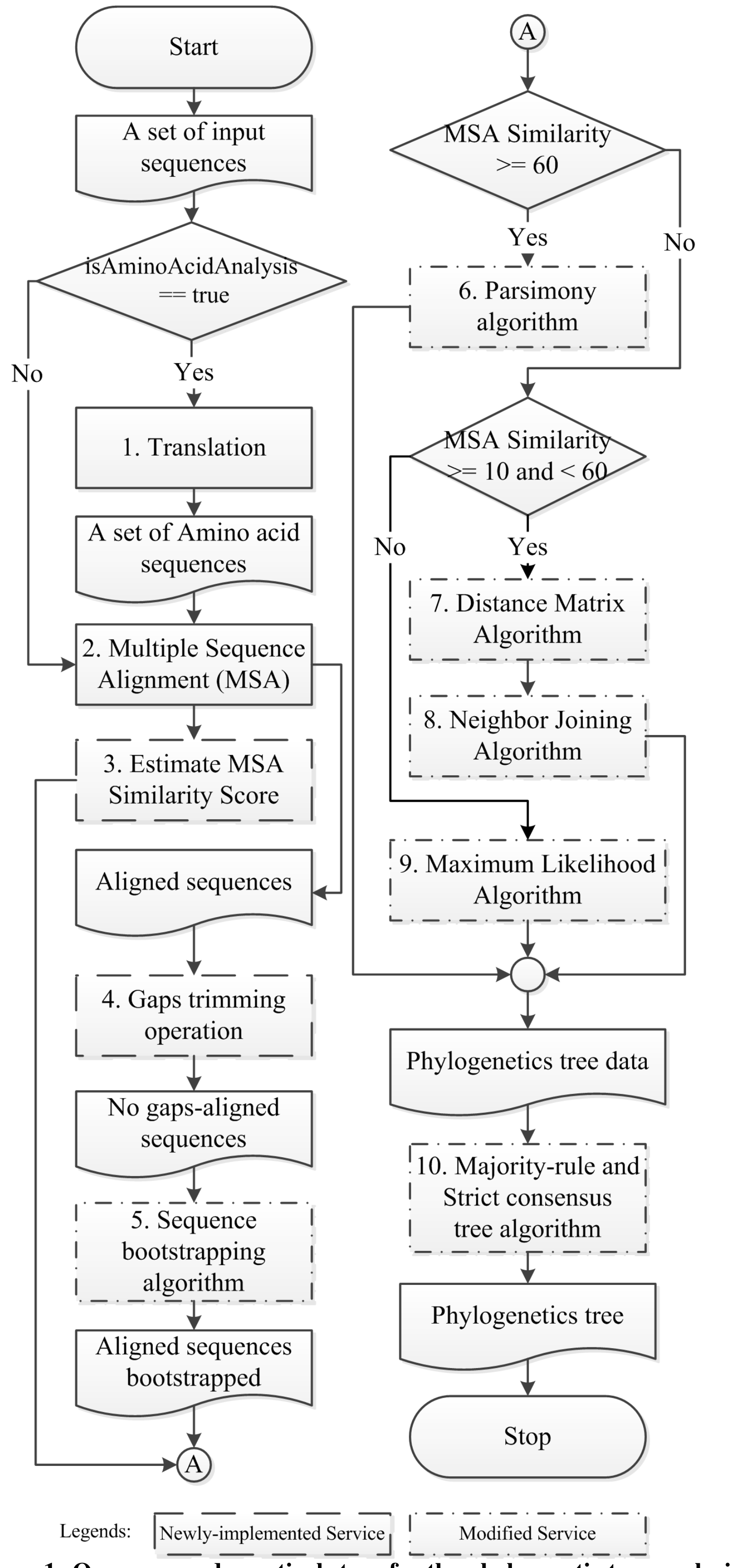

Start
A set of input sequences
isAminoAcidAnalysis == true
No
Yes
1. Translation
A set of Amino acid sequences
2. Multiple Sequence Alignment (MSA)
3. Estimate MSA Similarity Score
Aligned sequences
4. Gaps trimming operation
No gaps-aligned sequences
5. Sequence bootstrapping algorithm
Aligned sequences bootstrapped
A
A
MSA Similarity >= 60
Yes
No
6. Parsimony algorithm
MSA Similarity >= 10 and < 60
No
Yes
7. Distance Matrix Algorithm
8. Neighbor Joining Algorithm
9. Maximum Likelihood Algorithm
Phylogenetics tree data
10. Majority-rule and Strict consensus tree algorithm
Phylogenetics tree
Stop
Legends:
Newly-implemented Service
Modified Service

**Figure 1: Our proposed practical steps for the phylogenetic tree analysis**

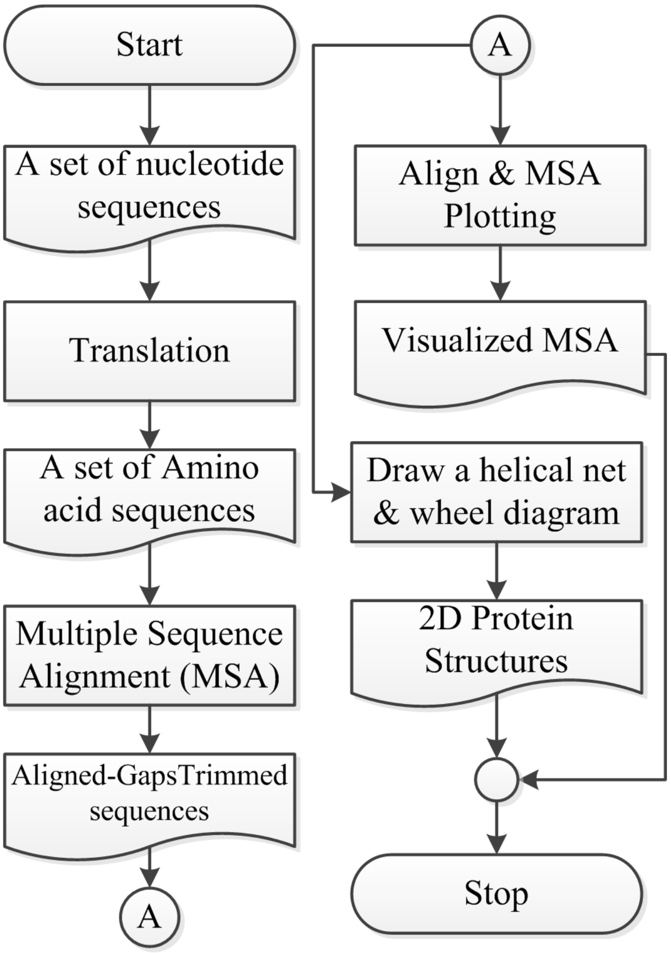

Start
A set of nucleotide sequences
Translation
A set of Amino acid sequences
Multiple Sequence Alignment (MSA)
Aligned-GapsTrimmed sequences
A
A
Align & MSA Plotting
Visualized MSA
Draw a helical net & wheel diagram
2D Protein Structures
Stop

**Figure 2: The proposed 2D protein structure analysis steps**

## 2.3 Annotation Support

Our workflow supports annotating information. The first one is 2D-protein structures generated from translated amino acid sequences as shown in Figure 2. The workflow produces two structural types, namely a diagram of helical net and helical wheel. We also propose our sequence alignment and MSA visualization with pretty formatting web services [5]. For the phylogenetic tree generation, we utilize the majority-rule and strict consensus tree algorithm [1][2] to draw the tree with the highest possibility.

## 2.4 The Preliminary Integrated Workflow

Our proposed analysis steps utilize web services provided by EMBL-EBI *(ClustalOmega)* and SIB (the rest). Table 1 shows all relevant web services and their brief descriptions. There are two types of services recommended for interfacing with Taverna which are the SOAP wrappers and strongly typed WSDL services. The preliminary integrated workflow is shown in Figure 3 with a collapsed and nested display. Then, all services have been composed with the strongly typed WSDL orientation. Table 2 shows the relationship between the output size and execu-

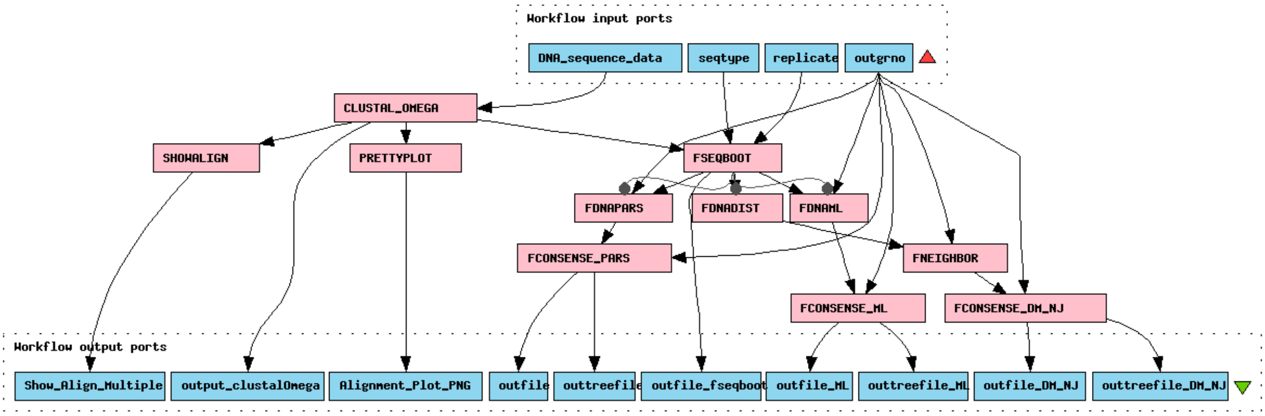


**Figure 3: The preliminary integrated workflow of tree inferring algorithms in a collapsed display**

**Table 1: Relevant web services for the proposed workflow**

| Process | Web Service | Description |
|---|---|---|
| Translation | transeq | Translate nucleic acid sequences |
| MSA | ClustalOmega | Seeded guide trees and HMM profile-profile techniques to generate alignments |
| | emma | Multiple sequence alignment (ClustalW wrapper) |
| Bootstrapping | fseqboot | Bootstrapped sequences algorithm |
| Tree Inferrence | fdnapars/fprotpars | DNA/Protein parsimony algorithm |
| | fdnadist/fprodist | DNA/Protein distance matrix algorithm |
| | fdnaml/fproml | DNA/Protein phylogenetic by maximum likelihood |
| | fneighbor | Phylogenies from distance matrix by N-J or UPGMA method |
| 2D Structure Generation | pepnet/pepwheel | Draw a helical net/wheel for a protein sequence |
| MSA Plot | prettyPlot | Draw a sequence alignment with pretty formatting |
| Alignment Plot | showalign | Display a multiple sequence alignment |
| Tree Generation | fconsense | Majority-rule and strict consensus tree algorithm |

tion time of the workflow. The experimental results show that when using 150 replicates, the workflow execution fails after being executed for 14.5 minutes. Our inspection reveals that the dataflow size has been exceeded in the bootstrapping step and this has induced errors in downstream processes.

**Table 2: The execution results of the preliminary tree inferring workflow for DNA of the 13-CDS dataset**

| No. of Replicates | Output Size (MB) | Execution Time (min.) |
|---|---|---|
| 10 | 3.4 | 14.3 |
| 20 | 5.1 | 15.3 |
| 50 | 10.2 | 17.8 |
| 100 | 18.6 | 35.4 |
| 150 | Failed: Dataflow exceeded | 14.5 |
| 200 | Failed: Dataflow exceeded | 23.4 |

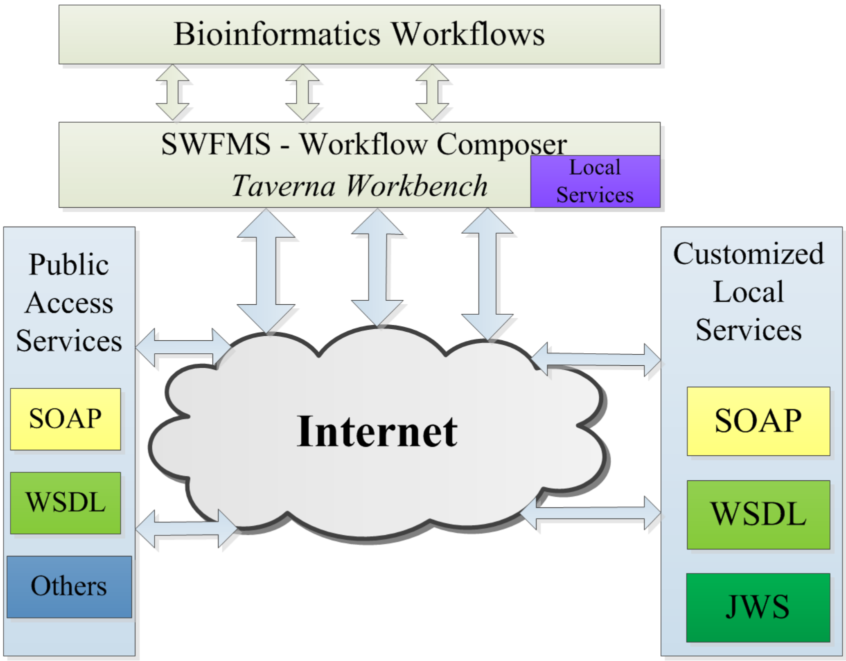


**Figure 4: The proposed architecture for better locality exploitation for the Taverva workflow environment**

## 3 Proposed Architecture

According to the performance results of the workflow in Table 2, we have found that the problem does not only occur at the client-side. At the SIB server-side, it sometimes lacks of device spaces on the server, even though we have composed the workflow using the strongly typed WSDL services in order to inform the server to clear no-use outputs. Therefore, in this paper, we propose solutions for solving the datalink limitation in the workflow using modified local web services. The services will produce and feed file references instead of directing the dataflow into the datalinks. Consequently, it will reduce the execution time, and will be able to support the number of replicates up to the recommendation for general practice. In addition, we propose a local helper workflow to identify the full file system path of each I/O service to be sent to the related downstream datalink.

**Table 3: The modified local services and their configuration**

| Local Service | Parameter | Input File Suffix | Output file Suffix |
|---|---|---|---|
| fseqboot | - | Direct aligned seqs | _outfile |
| fdnapars<br>fprotpars<br>fdnaml<br>fproml | fileref<br>auto<br>outgrno | [fseqboot]_outfile | _outfile<br>_outtreefile.treefile |
| fdnadist<br>fprotdist | fileref<br>auto | [fseqboot]_outfile | _outfile |
| fneighbor | fileref<br>outgrno | [fdnadist/fprodist]_outfile | _outfile<br>_outtreefile.treefile |
| fconsense | fileref<br>outgrno | _outtreefile.treefile | _outfile<br>_outtreefile.treefile |

### 3.1 Local Services Architecture

In this paper, we proposed local services architecture for the bioinformatics workflow development environment. The architecture is shown in Figure 4. Our system provides three types of services consisting of the SOAP service, WSDL service, and Java Web Service (JWS). We also utilize the built-in local services from Taverna for local I/O operations. The PHYLIPNEW command-line applications for tree inferring are wrapped using the Soaplab2 [26][27] to enable the applications to act like web services, both SOAP and WSDL are supported. These applications are shown in Table 3. The PHYLIPNEW version 3.69 and EMBOSS package version 6.4.0 [2] have been compiled and utilized for the system. Our Java method implementation is provided in JWS for estimating the MSA similarity score and gaps-trimming of the MSA result as we have described in the previous section. The services have been utilized by the Apache Axis2, version 1.6.3 that provides WSDL endpoints of Plain Old Java Objects (POJOs) [28] to the SWFMS, Taverna Workbench in our proposed architecture.

### 3.2 Local Services Configuration

Table 3 shows the tree inferring-related local services and their parameters customized details. The *fileref* parameter is a local file reference. We have investigated and extracted the I/O file name suffixes for each service. Then, bundled ACD (Ajax Command Definitions) [2] metadata files from the EMBOSS PHYLIPNEW applications have been modified. The ACD files describe command-line programs' behaviors and their parameters using the EMBOSS format. The direct data input of the services has been changed into a file reference. An example of the modified ACD files is shown below. Another parameter is the *auto*. It is used for the command-line prompts suppression.

```
string: fseqbootfile [
   parameter: “Y”
   template: “-sequence $$”
   help: “File containing one or more sequence alignments”
]

boolean: auto [
   additional: “Y”
   information: “Turn off prompts”
   default: “Y”
]
```

All local services have been deployed at our portal `http://bioservices.sci.psu.ac.th`. The bootstrapping workflow has been validated and measured with a series number of the bootstrapping replication within the range of 1,000 - 10,000. The performance tests are shown in Table 4. We have found that the practical numbers of replicates are 1,000 and 2,000 and they generate 196 MB and 340 MB outputs respectively. In addition, we manually run the downstream tree inferring steps using the command-line programs with these bootstrapped results. Their outputs are just less than 2 MB.

### 3.3 Integrated Automatic Workflow and Its Performance

We re-compose the integrated tree inferring workflow using our deployed local and helper services as shown in the collapsed style in Figure 5. The expanded and nested workflow style is also attached as a supplement file. Our integrated workflow is available at `http://www.myexperiment.org/workflows/4945.html`.

Table 5 shows execution times of the integrated tree inferring workflow for DNA using our own deployed local and helper services. In practical experiments, it is widely accepted to use 1,000 replicates in the bootstrapping step. Our integrated workflow consumes execution time less than two hours for the 1,000 replicates. Currently, there is no other public MSA gaps-trimming service or workflow for trimming the result yet. Work is on progress to improve the algorithms to trim gaps for more types of datasets. Figure 6 depicts phylogenetic trees for the PARS (a), DIST-NJ (b), and ML algorithms (c) respectively. The trees confirm that our implemented workflows are validated and the results are satisfied by our bioinformaticians.

**Table 4: The performance tests of the bootstrapping workflow for DNA of the 13-CDS dataset using local services**

| No. of Replicates | Output Size (MB) | Execution Time (min.) |
|---|---|---|
| 1,000 | 196 | 0.4 |
| 2,000 | 340 | 0.9 |
| 3,000 | 509 | 1.8 |
| 4,000 | 679 | 2.9 |
| 5,000 | 848 | 4.4 |
| 10,000 | 1,700 | 5.9 |

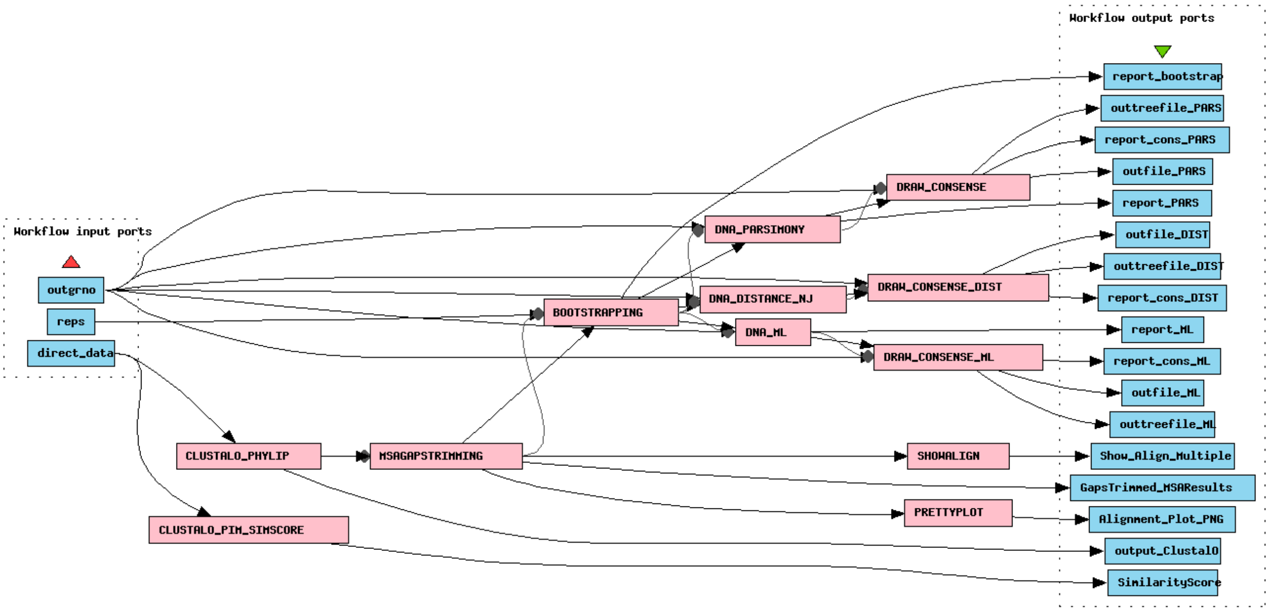


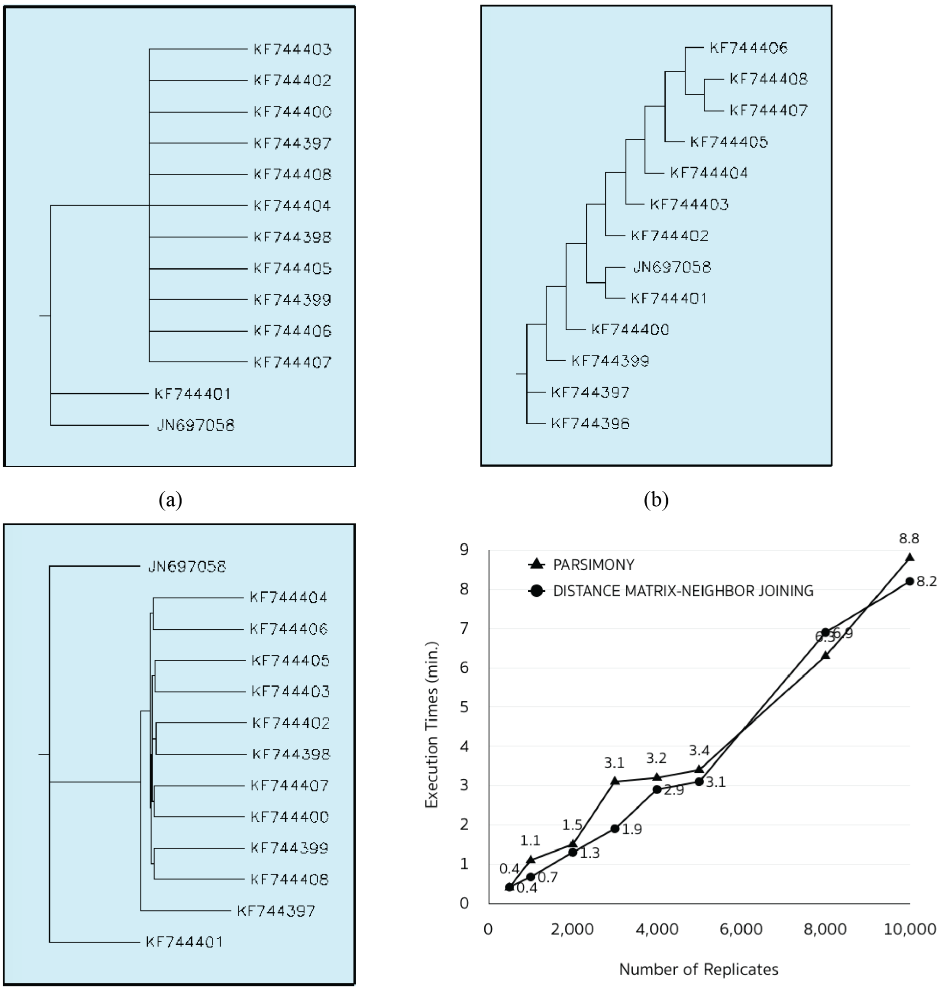
**Figure 5: The integrated workflow of tree inferring algorithms using local and helper services in a collapsed display**

# 4 Discussion

This section discusses the workflow composition and performance, and the limitation of Taverna workbench.

**Table 5: The execution times of the integrated tree inferring workflow for DNA using our own deployed local and helper services**

| No. of Replicates | Execution Time (hrs.) |
|---|---|
| 500 | 1.1 |
| 1,000 | 1.7 |
| 2,000 | 3.3 |



(a)

(b)

(c)

(d)

**Figure 6: The phylogenetic tree of PARS (a), DIST-NJ (b), ML (c) and the comparison of the PARS and DIST-NJ algorithms in terms of relationship between the execution time and the number of replicates (d).**

## 4.1 Workflow Performance and Limitation

From our experiments, tree inferring steps are both data and computational intensive. The bootstrapping step is the most data intensive, whereas the ML algorithm is the most computational intensive. Figure 6 (d) shows the comparison of the PARS and the DIST-NJ algorithms of which their execution times include the bootstrapping step. It takes less than ten minutes for inferring a tree with 10,000 replicates. On the other hand, it takes a few hours to run the tree inferring step using the ML algorithm. In practice, it is difficult to choose which is the most suitable number of replicates in the bootstrapping step [7]. For example, at some points, a tree may not change its attributes anymore, even though we increase the number of replicates. For this workflow, we employ the least number of 1,000 replicates because it is widely used and accepted. Our proposed architecture produces a rather high throughput, not a high performance computing (HPC) framework. However, the accuracy of the MSA is a critical term in phylogenetic analysis that may induce biases in the tree topology and its branch lengths.

### 4.2 Limitation of the Taverna Workbench

We have found that if a workflow is very complicated, Taverna may induce an out-of-memory error, and then may often be in a no-response state. A complicated workflow generally redraws its graphical elements when Taverna has detected some changes affecting the workflow such as adding and removing a service or any object. This consumes a large amount of main memory and tends to lead Taverna to a no-response state.

In addition, the Taverna Workbench does not support intermediate results tracing. In case of a complex integrated workflow, e.g. the integrated workflow in this paper. If the nested workflow has completed its job, it may not be able to trace and inspect any intermediate results, e.g. the result of MSA gaps removal. The users have to wait until all nested workflows have already completed their jobs before he is able to inspect the results. Therefore, it is difficulty to debug a large workflow. However, myGrid introduces Taverna Server [10] for workflow invocation and user interaction processors handling. The BioVeL Project [29] is an example portal for biodiversity study [12] that offers tool suites for robust workflow running. Moreover, users with no experiences with SWFMS may not be able to easily compose any scientific workflow in a practical scenario using relevant web services due to lack of knowledge in SOAP and WSDL web services for the scientific workflow composition.

## 5 Conclusion and Outlook

In this paper, we propose an integrated automatic workflow for the phylogenetic tree analysis based on the MSA similarity score using the public access web services and our own deployed local web services of the PHYLIPNEW package. The workflow provides preprocessing data, tree inferring algorithms, and annotation support. The MSA similarity score is estimated from the PIM data using our proposed equation. The workflow supports popular-tree inferring such as the PARS, DIST-NJ, and ML algorithms. The workflow has been validated, its performances have been measured, and its results have been verified. Our bioinformaticians are satisfied by the results. This paper proposes the new integrated automatic workflow which will be beneficial to bioinformaticians with an intermediate level of knowledge and experiences. Our work is on progress for enhancing workflow invocation to support smart execution as well as Workflow-as-a-Service (WaaS) [10][11].

We have been investigating to improve features of the workflow in order to have aspects for the phylogenetic analysis, such as data partitioning, evolutionary model test [30][31], and probabilistic models of inferring tree, e.g. Bayesian approach [29]. In addition, we will utilized another web service for MSA tool included in the workflow as alternative options to the user, e.g. MAFFT.

## Download

All our local services can be accessed at the portal `http://bioservices.sci.psu.ac.th`. The integrated workflow is available at `http://www.myexperiment.org/workflows/4945.html`.

## Competing interests

The authors declare that they have no competing interests.

## Acknowledgements

This work is funded by the INTACT, Department of Information and Communication Technology, Faculty of Science, Prince of Songkla University, Thailand.

## References

[1] J. Felsenstein. PHYLIP - Phylogeny inference package (version 3.2). *Cladistics*, 5:164–166, 1989.

[2] P. Rice, I. Longden and A. Bleasby. EMBOSS: The European Molecular Biology Open Software Suite (2000). *Trends in Genetics*, 16(6):276–277, 2000.

[3] K. Tamura, G. Stecher, D. Peterson, A. Filipski and S. Kumar. MEGA6: Molecular Evolutionary Genetics Analysis version 6.0. *Molecular Biology and Evolution*, 30:2725–2729, 2013.

[4] W. Li, A. Cowley, M. Uludag, T. Gur, H. McWilliam, S. Squizzato, Y. M. Park, N. Buso and R. Lopez. The EMBL-EBI bioinformatics web and programmatic tools framework. *Nucleic Acids Research*, 43:W580–W584, 2015.

[5] M. Pagni, J. Hau and H. Stockinger. A Multi-protocol Bioinformatics Web Service: Use SOAP, Take a REST or Go with HTML. In *Proc. IEEE International Symposium on Cluster Computing and the Grid*, pages 728–734. Lyon, France, 2008.

[6] L. J. Revell and S. A. Chamberlain. Rphylip: an R interface for PHYLIP. *Methods in Ecology and Evolution*, 5:976–981, 2014.

[7] A. L. Bazinet, D. J. Zwickl and M. P. Cummings. A Gateway for Phylogenetic Analysis Powered by Grid Computing Featuring GARLI 2.0. *Syst Biol*, 63(5):syu031v1–syu031, 2014.

[8] R. Sánchez, F. Serra, J. Tárraga et al. Phylemon 2.0: a suite of web-tools for molecular evolution, phylogenetics, phylogenomics and hypotheses testing. *Nucleic Acids Research*, 10(1093):1–5, 2011.

[9] S. Guindon, J.-F. Dufayard, V. Lefort, M. Anisimova, W. Hordijk and O. Gascuel. New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of PhyML 3.0. *Systematic Biology*, 59(3):307–321, 2010.

[10] K. Wolstencroft, R. Haines, D. Fellows et al. The Taverna workflow suite: designing and executing workflows of web services on the desktop web or in the cloud. *Nucleic Acids Research*, 41(Web Server issue):W557–W561, 2013.

[11] W. Tan, K. Chard, D. Sulakhe, R. Madduri, I. Foster, S. Soiland and C. Goble. Scientific workflows as services in caGrid: a Taverna and gRAVI approach. In *Proc. IEEE International Conference on Web Services*, pages 413–420. 2009.

[12] C. Mathew, A. Guntsch, M. Obst, S. Vicario, R. Haines, A. Williams, Y. de Jong and C. Goble. A semi-automated workflow for biodiversity data retrieval, cleaning, and quality control. *Biodiversity Data Journal*, 2:e4221, 2014.

[13] J. Ruiz, J. Garrido, J. Santander-Vela, S. Sanchez-Exposito and L. Verdes-Montenegro. Astrotaverna–building workflows with virtual observatory services. *Astronomy and Computing*, 7-8:3–11, 2014. Special Issue on The Virtual Observatory: I.

[14] P. Missier, S. Soiland-Reyes, S. Owen, W. Tan, A. Nenadic, I. Dunlop, A. Williams, T. Oinn and C. Goble. Taverna, reloaded. In *Proceedings of the 22$^{nd}$ International Conference on Scientific and Statistical Database Management*, SSDBM'10, pages 471–481. Springer-Verlag, Berlin, Heidelberg, 2010.

[15] I. Altintas, J. Wang, D. Crawl and W. Li. Challenges and approaches for distributed workflow-driven analysis of large-scale biological data. In *Proc. Workshop on Data analytics in the Cloud at EDBT/ICDT 2012 Conference*, pages 73–78. Berlin, Germany, 2012.

[16] Y. Zhao, Y. Li, I. Raicu, S. Lu, W. Tian and H. Liu. Enabling scalable scientific workflow management in the Cloud. *Future Generation Computer Systems*, 46(Issue C):3–16, 2015.

[17] Y. Zhao, Y. Li, I. Raicu, S. Lu, C. Lin, Y. Zhang, W. Tian and R. Xue. A service framework for scientific workflow management in the Cloud. *IEEE Transactions on Services Computing*, PP(99):1–14, 2014.

[18] Y. Zhao, Y. Li, I. Raicu, C. Lin, W. Tian and R. Xue. Migrating Scientific Workflow Management Systems from the Grid to the Cloud. *Cloud Computing for Data Intensive Applications*, pages 231–256, 2014.

[19] K. Damkliang, P. Tandayya, T. Phusantisampan and U. Sangket. Taverna workflow and supporting services for single nucleotide polymorphisms analysis. In *Proc. International Conference on Information Management and Engineering*, pages 27–31. Kuala Lumpur, Malaysia, 2009.

[20] T. Tatusova, S. Ciufo, B. Fedorov, K. O'Neill and I. Tolstoy. RefSeq microbial genomes database: new representation and annotation strategy. *Nucleic Acids Research*, 42(1):D553–D559, 2014.

[21] F. Sievers, A. Wilm, D. Dineen et al. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Molecular Systems Biology*, 7(539):1–6, 2011.

[22] T. Lassmann, O. Frings and E. L. L. Sonnhammer. Kalign2: high-performance multiple alignment of protein and nucleotide sequences allowing external features. *Nucleic Acids Research*, 37(3):858–865, 2009.

[23] K. Katoh and H. Toh. Recent developments in the MAFFT multiple sequence alignment program. *Briefings in Bioinformatics*, 9(4):286–298, 2008.

[24] R. C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–1797, 2004.

[25] B. P. Blackburne and S. Whelan. Measuring the distance between multiple sequence alignments. *BIOINFORMATICS*, 28(4):495–502, 2012.

[26] M. Senger, P. Rice and T. Oinn. Soaplab - a unified sesame door to analysis tools. In *Proc. UK e-Science All Hands Meeting 2003*, pages 509–513. Nottingham, UK, 2003.

[27] M. Senger, P. Rice, T. Oinn and M. Uludag. Soaplab2: more reliable sesame door to bioinformatics programs. In *Proc. The 9th annual Bioinformatics Open Source Conference*. Toronto, Canada, 2008.

[28] S. Perera, C. Herath, J. Ekanayake, E. Chinthaka, A. Ranabahu, D. Jayasinghe, S. Weerawarana and G. Daniels2. Axis2, middleware for next generation web services. In *Proc. IEEE International Conference on Web Services*, pages 833–840. Chicago, USA, 2006.

[29] S. Vicario, B. Balech, G. Donvito, P. Notarangelo and G. Pesole. The BioVel Project: Robust phylogenetic workflows running on the GRID. *EMBnet.journal*, 18:77–79, 2012.

[30] R. Lanfear, B. Calcott, S. Y. W. Ho and S. Guindon. Partitionfinder: Combined selection of partitioning schemes and substitution models for phylogenetic analyses. *Molecular Biology and Evolution*, 29(6):1695–1701, 2012.

[31] R. Lanfear, B. Calcott, D. Kainer, C. Mayer and A. Stamatakis. Selecting optimal partitioning schemes for phylogenomic datasets. *BMC Evolutionary Biology*, 14(1):1–14, 2014.