ZOOLOGICAL RESEARCH

AutoSeqMan: batch assembly of contigs for Sanger sequences

Jie-Qiong Jin[1], Yan-Bo Sun[1,*]

[1] *State Key Laboratory of Genetic Resources and Evolution, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming Yunnan* 650223*, China*

ABSTRACT

With the wide application of DNA sequencing technology, DNA sequences are still increasingly generated through the Sanger sequencing platform. SeqMan (in the LaserGene package) is an excellent program with an easy-to-use graphical user interface (GUI) employed to assemble Sanger sequences into contigs. However, with increasing data size, larger sample sets and more sequenced loci make contig assemble complicated due to the considerable number of manual operations required to run SeqMan. Here, we present the 'autoSeqMan' software program, which can automatedly assemble contigs using SeqMan scripting language. There are two main modules available, namely, 'Classification' and 'Assembly'. Classification first undertakes preprocessing work, whereas Assembly generates a SeqMan script to consecutively assemble contigs for the classified files. Through comparison with manual operation, we showed that autoSeqMan saved substantial time in the preprocessing and assembly of Sanger sequences. We hope this tool will be useful for those with large sample sets to analyze, but with little programming experience. It is freely available at https://github.com/Sun-Yanbo/autoSeqMan.

Keywords: Batch processing; Sanger sequences; Contig assembly; SeqMan

INTRODUCTION

DNA sequencing technology has experienced a revolutionary shift from automated Sanger sequencing (Sanger et al., 1977) to next-generation sequencing (NGS; reviewed by Shendure & Ji (2008) and Shendure et al. (2004)). Although NGS has dominated due to its high throughput (Schuster, 2008), it is not suitable for many population studies due to high costs and other limiting factors. For example, errors are always introduced in final assembly and/or annotation results using NGS data (Bickhart et al., 2017), and thus variations detected in high-throughput analyses require validation by Sanger sequencing (Wall et al., 2014). Furthermore, for some present population genomic studies, error rates have been found to increase with increasing depth of coverage for Illumina data, and thus caution is needed when interpreting the results of next-generation sequencing-based association studies (Wall et al., 2014). As such, Sanger sequencing technology is still widely used in many research fields, including evolutionary taxonomy based on short DNA sequences (Chen et al., 2017), evolutionary history study of wild animals (Yuan et al., 2016), biodiversity estimates and influencing factors (Zhou et al., 2017), and validation of mutations identified from high-throughput analyses (Sun et al., 2013).

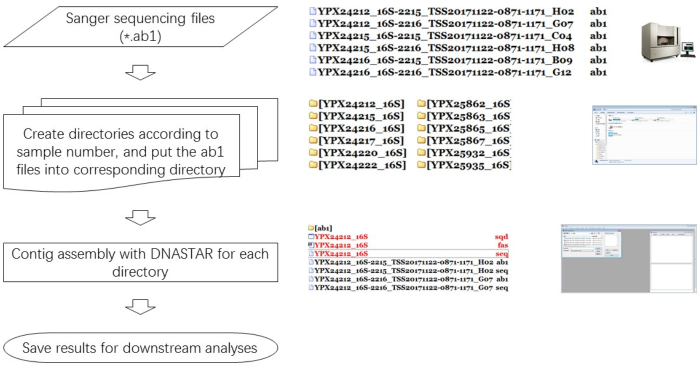
Further, with the wide application of DNA sequencing technology, e.g. DNA barcoding, which uses short and standardized DNA sequences for individual identification of organisms (Hajibabaei et al., 2007; Savolainen et al., 2005), Sanger sequencing data are continuing to be accumulated among evolutionary taxonomists and others. Thus, batch manipulation of these Sanger sequences has become an important task before downstream analyses, especially for those who doesn't have programming or bioinformatics background or experiments. Although several sequence manipulation packages for general purpose issues have been published previously, including MEGA (Kumar et al., 2016), EMBOSS (Rice et al., 2000), and FasParser (Sun, 2017), these packages are all based on assembled contigs (a consensus

Received: 21 December 2017; Accepted: 08 February 2018

Foundation items: The development of this package was supported by the National Natural Science Foundation of China (31671326) and the Youth Innovation Promotion Association, Chinese Academy of Sciences

*Corresponding author, E-mail: sunyanbo@mail.kiz.ac.cn

DOI: 10.24272/j.issn.2095-8137.2018.027


Sanger sequencing files
(*.ab1)
Create directories according to sample number, and put the ab1 files into corresponding directory
Contig assembly with DNASTAR for each directory
Save results for downstream analyses

```
YPX24212_16S-2215_TSS20171122-0871-1171_H02    ab1
YPX24212_16S-2216_TSS20171122-0871-1171_G07    ab1
YPX24215_16S-2215_TSS20171122-0871-1171_C04    ab1
YPX24215_16S-2216_TSS20171122-0871-1171_H08    ab1
YPX24216_16S-2215_TSS20171122-0871-1171_B09    ab1
YPX24216_16S-2216_TSS20171122-0871-1171_G12    ab1

[YPX24212_16S]   [YPX25862_16S
[YPX24215_16S]   [YPX25863_16S
[YPX24216_16S]   [YPX25865_16S
[YPX24217_16S]   [YPX25867_16S
[YPX24220_16S]   [YPX25932_16S
[YPX24222_16S]   [YPX25935_16S]

[ab1]
YPX24212_16S                                       sqd
YPX24212_16S                                       fas
YPX24212_16S                                       seq
YPX24212_16S-2215_TSS20171122-0871-1171_H02 ab1
YPX24212_16S-2215_TSS20171122-0871-1171_H02 seq
YPX24212_16S-2216_TSS20171122-0871-1171_G07 ab1
YPX24212_16S-2216_TSS20171122-0871-1171_G07 seq
```

Figure 1 Overview of assembling tasks for Sanger sequences

Common tasks for assembling Sanger sequences include (1) classifying sequence files into corresponding folders (always completed manually) and (2) opening SeqMan to add the ab1 files that belong to a same sample and then assembling them (needing considerable mouse operations). The autoSeqMan program was designed based on these steps, with the 'Classification' and 'Assembly' functions corresponding to step 2 and 3, respectively.

region of overlapping DNA segments) and no key consideration has been taken on the batch assembly of Sanger sequences.

SeqMan is a popular program in the LaserGene software package (DNAStar, Inc., Madison, WI, USA), which is used for assembling Sanger sequences into contigs and has been widely applied in a great number of studies. It can handle two to thousands of Sanger sequences at one time but requires a considerable number of manual operations (e.g., mouse actions, Figure 1) to run. Hence, it is complicated and time-expensive for those with large sets of samples to assemble. Fortunately, since the release of Version 7, SeqMan now provides a scripting language, including commands for opening, naming, saving, and closing projects, and a single script may be used to execute multiple assemblies consecutively without manual intervention.

Here, we developed a program called autoSeqMan, which provides a simple way to automatedly classify Sanger sequences and then consecutively assemble them on a personal computer. It is mainly designed for researchers with large sets of samples with one or more loci sequenced.

IMPLEMENTATION AND REQUIREMENTS

autoSeqMan was developed into a standalone Windows desktop application (compiled and tested in Windows 7/10). It involves two modules, 'Classification' and 'Assembly', corresponding to steps 2 and 3 in Figure 1, respectively. Each module can handle multiple files and needs the user to select the directory either containing the raw Sanger sequence files (*.ab1 files) or containing the classified sub-folders created by 'Classification'. Theoretically, there is no limit to the number of files that can be analyzed.

This tool requires the sequence files be named in a specialized format, in which the sample ID should be present at the beginning of the file name. The Classification module will recognize the sample ID by the appropriate delimiter and then create sub-folders (see below). For convenience, autoSeqMan also provides a "Rename" tool to help users rename the ab1 files for the below analyses.

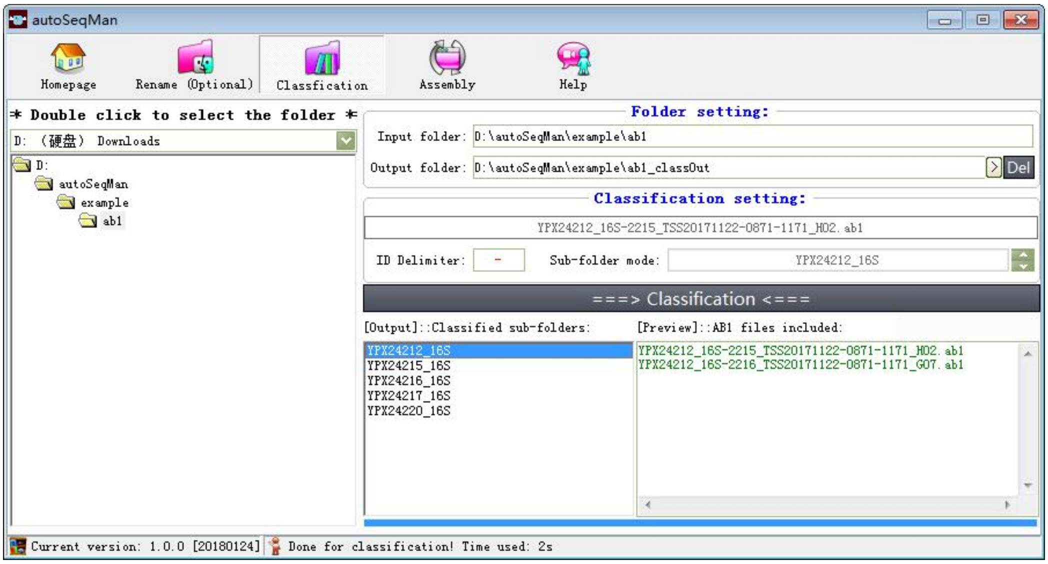
CLASSIFICATION

This function is designed to automatedly create sub-folders according to the sample ID and/or sequenced locus. All downstream analyses are performed in the corresponding sub-folders, where all analyzed results are also saved. According to our laboratory experience, this is an efficient and convenient way to manage and query laboratory samples (Chen et al., 2017; Zhou et al., 2017).

The only input is the directory name, which contains the raw ab1 files. There are several input prerequisites required for Classification performance. First, all files must be stored in a same directory. Second, all files must be named according to a certain pattern, i.e., "sample-locus-others". For example, the file name "YPX24212_16S-2215_TSS 20171122-0871-1171_H02.ab1" denotes that it is a DNA sequence of 16S and the sample number is "YPX24212". The program will automatedly recognize the filename according to the user-specified delimiter and then create a sub-folder "YPX24212_16S" in the main output folder. The delimiter can be "-", "_", or other. After classification, the program will list all sub-folders, and the user can look at the files classified into each sub-folder by simply clicking the folder name (Figure 2).

ASSEMBLY

This function will automatedly assemble the classified sequence files. It will first read the list of classified sub-folders created by the 'Classification' function, and then generate a SeqMan script for consecutively assembling the sequences in each sub-folder. To perform this function, the user must first install the LaserGene package (version 7 or higher), and then tell autoSeqMan the full path of the SeqMan program (which can be always recognized automatedly by autoSeqMan), after which the program will complete all assembly tasks and save the assembly results automatedly. The default script will generate all SQD, FAS, and SEQ results (Figure 3).



* Double click to select the folder *

D: (硬盘) Downloads

D:
autoSeqMan
example
ab1

Folder setting:

Input folder: D:\autoSeqMan\example\ab1

Output folder: D:\autoSeqMan\example\ab1_classOut

Classification setting:

YPX24212_16S-2215_TSS20171122-0871-1171_H02.ab1

ID Delimiter: -   Sub-folder mode: YPX24212_16S

[Output]::Classified sub-folders:

```
YPX24212_16S
YPX24215_16S
YPX24216_16S
YPX24217_16S
YPX24220_16S
```

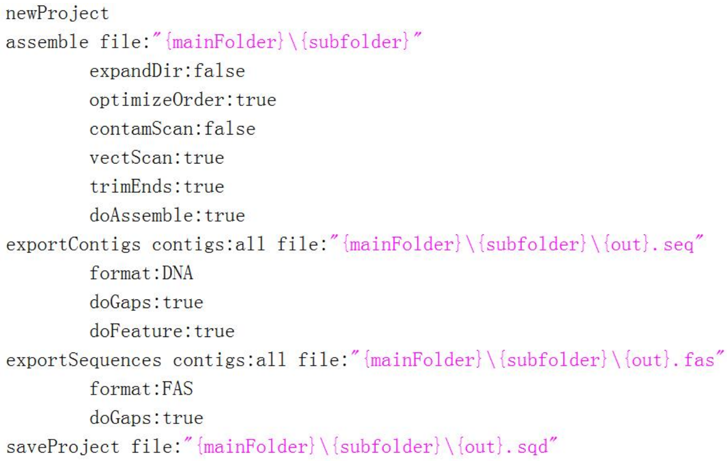
[Preview]::AB1 files included:

```
YPX24212_16S-2215_TSS20171122-0871-1171_H02.ab1
YPX24212_16S-2216_TSS20171122-0871-1171_G07.ab1
```

Current version: 1.0.0 [20180124]   Done for classification! Time used: 2s

Figure 2 Overview of ‘Classification’ function in autoSeqMan

To perform this function, user needs to tell autoSeqMan the full path of a directory which contains all raw ab1 files. The sample name should be present at the beginning of the file name, which will be recognized and extracted by the autoSeqMan according to the specified delimiter.



```
newProject
assemble file:"{mainFolder}\{subfolder}"
        expandDir:false
        optimizeOrder:true
        contamScan:false
        vectScan:true
        trimEnds:true
        doAssemble:true
exportContigs contigs:all file:"{mainFolder}\{subfolder}\{out}.seq"
        format:DNA
        doGaps:true
        doFeature:true
exportSequences contigs:all file:"{mainFolder}\{subfolder}\{out}.fas"
        format:FAS
        doGaps:true
saveProject file:"{mainFolder}\{subfolder}\{out}.sqd"
```

Figure 3 Default SeqMan script for assembling Sanger sequences

This script will be generated automated by autoSeqMan. There is no need for users to edit it. In default, all SQD, SEQ, and FAS outputs are generated.

PERFORMANCE

The main aim of autoSeqMan is to save manual operation in preparing files and running the SeqMan program. To evaluate its performance, we applied this tool to our laboratory data (Chen et al., 2017; Zhou et al., 2017). In this test, one hundred samples were used, each of which had two ab1 files available. Results showed that the Classification operation created sub-folders (named sample ID as well as locus name if provided) and moved the appropriate files into the sub-folders within 8 s, substantially less than the time used for manual operation (about 1 h, as tested by our colleagues). Performance of the Assembly operation greatly depended on the running efficiency of SeqMan. In this test, the Assembly module required 64 s to consecutively assembly contigs for the classified sequences, also substantially less than the ~2 h required for manual operation, suggesting the significance of autoSeqMan in dealing with large date sets.

LIMITATIONS

It is important to note that autoSeqMan does not undertake any filtration manipulation on the sequence data, even though poor-quality sequence ends are always present. Thus, after running autoSeqMan, users should undertake quality control measures of the final assembly with SeqMan. In addition, the output Fasta files will have very long IDs, which might introduce some errors in subsequent sequence analyses. If necessary, users can use the “Sort & Rename” function of FasParser (Sun, 2017) to shorten these IDs.

COMPETING INTERESTS

The authors declare that they have no competing interests.

AUTHORS’ CONTRIBUTIONS

Y.B.S. and J.Q.J. designed the study and wrote the manuscript. Y.B.S. wrote the software. J.Q.J. evaluated the performance of autoSeqMan. All authors read and approved the final manuscript.

ACKNOWLEDGEMENTS

Special thanks to our colleagues, Hong-Man Chen, Yu-Qi He, and Fang Yan, for their helpful suggestions on the improvement of autoSeqMan.

REFERENCES

Bickhart DM, Rosen BD, Koren S, Sayre BL, Hastie AR, Chan S, Lee J, Lam ET, Liachko I, Sullivan ST, Burton JN, Huson HJ, Nystrom JC, Kelley CM, Hutchison JL, Zhou Y, Sun JJ, Crisà A, De León FAP, Schwartz JC, Hammond JA, Waldbieser GC, Schroeder SG, Liu GE, Dunham MJ, Shendure J, Sonstegard TS, Phillippy AM, Van Tassell CP, Smith TP. 2017. Single-molecule sequencing and chromatin conformation capture enable *de novo* reference assembly of the domestic goat genome. *Nature Genetics,* **49**(4): 643–650.

Chen JM, Zhou WW, Poyarkov NA, Jr., Stuart BL, Brown RM, Lathrop A, Wang YY, Yuan ZY, Jiang K, Hou M, Chen HM, Suwannapoom C, Nguyen SN, van Duong T, Papenfuss TJ, Murphy RW, Zhang YP, Che J. 2017. A novel multilocus phylogenetic estimation reveals unrecognized diversity in Asian horned toads, genus *Megophrys sensu lato* (Anura: Megophryidae). *Molecular Phylogenetics & Evolution,* **106**: 28–43.

Hajibabaei M, Singer GA, Hebert PD, Hickey DA. 2007. DNA barcoding: how it complements taxonomy, molecular phylogenetics and population genetics. *Trends in Genetics,* **23**(4): 167–172.

Kumar S, Stecher G, Tamura K. 2016. MEGA7: Molecular evolutionary genetics analysis version 7.0 for bigger datasets. *Molecular Biology and Evolution,* **33**(7): 1870–1874.

Rice P, Longden I, Bleasby A. 2000. EMBOSS: the european molecular biology open software suite. *Trends in Genetics,* **16**(6): 276–277.

Sanger F, Nicklen S, Coulson AR. 1977. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America,* **74**(12): 5463–5467.

Savolainen V, Cowan RS, Vogler AP, Roderick GK, Lane R. 2005. Towards writing the encyclopedia of life: an introduction to DNA barcoding. *Philosophical Transactions of the Royal Society B: Biological Sciences,* **360**(1462): 1805–1811.

Schuster SC. 2008. Next-generation sequencing transforms today’s biology. *Nature Methods,* **5**(1): 16–18.

Shendure J, Ji H. 2008. Next-generation DNA sequencing. *Nature Biotechnology,* **26**(10): 1135–1145.

Shendure J, Mitra RD, Varma C, Church GM. 2004. Advanced sequencing technologies: methods and goals. *Nature Reviews Genetics,* **5**(5): 335–344.

Sun YB. 2017. FasParser: a package for manipulating sequence data. *Zoological Research,* **38**(2): 110–112.

Sun YB, Zhou WP, Liu HQ, Irwin DM, Shen YY, Zhang YP. 2013. Genome-wide scans for candidate genes involved in the aquatic adaptation of dolphins. *Genome Biology and Evolution,* **5**(1): 130–139.

Wall JD, Tang LF, Zerbe B, Kvale MN, Kwok PY, Schaefer C, Risch N. 2014. Estimating genotype error rates from high-coverage next-generation sequence data. *Genome Research,* **24**(11): 1734–1739.

Yuan ZY, Zhou WW, Chen X, Poyarkov NA, Jr., Chen HM, Jang-Liaw NH, Chou WH, Matzke NJ, Iizuka K, Min MS, Kuzmin SL, Zhang YP, Cannatella DC, Hillis DM, Che J. 2016. Spatiotemporal diversification of the true frogs (*Genus Rana*): A historical framework for a widely studied group of model organisms. *Systematic Biology,* **65**(5): 824–842.

Zhou WW, Jin JQ, Wu J, Chen HM, Yang JX, Murphy RW, Che J. 2017. Mountains too high and valleys too deep drive population structuring and demographics in a Qinghai-Tibetan Plateau frog *Nanorana pleskei* (Dicroglossidae). *Ecology and Evolution,* **7**(1): 240–252.