

**Title**

Author Last and first names

Address

# Abstract

Content of abstract.

# Introduction

- BWA (Li and Durbin 2009)
- Minimap2 (Li 2018)
- SPAdes (Bankevich et al. 2012)
- Flye (Kolmogorov et al. 2019)
- FMLRC (Wang et al. 2018)
- GeSeq (Tillich et al. 2017)
- GetOrganelle (Jin et al. 2020)
- ptGAUL (Zhou et al. 2023)
- CLAW (Phillips et al. 2024)
- PMAT (Bi et al. 2024)
- Oatk (Zhou et al. 2024)
- TIPPo (Xian et al. 2025)
- Subsampling (Efron 1987)
- NextDenovo (Hu et al. 2024)

# Materials and Methods

(Table ??)

(Figure 1)

- Flye (Kolmogorov et al. 2019)
- JellyFish (Marçais and Kingsford 2011)
- BLAST (Altschul et al. 1997)
- SeqKit (Shen et al. 2016)
- MAFFT (Katoh and Standley 2013)

(Figure 1)

- Bandage (Wick et al. 2015)
- Canu (Koren et al. 2017)
- NextDenovo better than Canu (Wick and Holt 2021)

(Table ??)

(Table S2)

# Results

## Comparison with other plastid assembly pipelines

(Table ??)

(Supporting Materials – Plastid genome assemblies using the six pipelines)

(Table S4)

(Table S5)

## Subsampling-based plastome assemblies

(Table ??)

(Table ??)

(Table ??)

## Three-stage of subsampling-based assembly

- (Table ??)

- (Table ??)
- (Table ??)

## Discussion

Polap (Plant Organelle Long-read Assembly Pipeline v0.4.3.7), which includes the subsampling-based plastid genome assembly feature, is available under the GNU General Public License version 3.0 at <http://github.com/goshng/polap>.

## Supplementary Material

Supplementary material, including 10 tables and three figures, is appended to the main text of this manuscript. A BASH script for executing the pipeline used to generate the results presented in the manuscript is also included.

## Acknowledgements

We thank Jeffrey L. Thorne for improving the presentation of this work.

## Author Contributions

S.C.C. developed the Polap pipeline and prepared the manuscript.

## Conflict of Interest

The author declare no conflicts.

## Data availability

Polap (Plant Organelle Long-read Assembly Pipeline v0.4.3.7) is available under the GNU General Public License version 3.0 at <http://github.com/goshng/polap>. The results presented in this manuscript are available at Figshare: <https://figshare.com/s/ec1cb394870c7727a2d4>.

## References

Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* 25:3389–3402.

<https://doi.org/10.1093/nar/25.17.3389>

Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, Lesin VM, Nikolenko SI, Pham S, Prjibelski AD, et al. 2012. SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *J Comput Biol.* 19:455–477. <https://www.liebertpub.com/doi/abs/10.1089/cmb.2012.0021>

Bi C, Shen F, Han F, Qu Y, Hou J, Xu K, Xu L, He W, Wu Z, Yin T. 2024. PMAT: An efficient plant mitogenome assembly toolkit using low-coverage HiFi sequencing data. *Hortic Res.* 11:uhae023. <https://doi.org/10.1093/hr/uhae023>

Efron B. 1987. Better bootstrap confidence intervals. *J Amer Statistical Assoc.* 82:171–185.  
<https://www.tandfonline.com/doi/abs/10.1080/01621459.1987.10478410>

Hu J, Wang Z, Sun Z, Hu B, Ayoola AO, Liang F, Li J, Sandoval JR, Cooper DN, Ye K, et al. 2024. NextDenovo: An efficient error correction and accurate assembly tool for noisy long reads. *Genome Biol.* 25:107.  
<https://doi.org/10.1186/s13059-024-03252-4>

Jin J-J, Yu W-B, Yang J-B, Song Y, dePamphilis CW, Yi T-S, Li D-Z. 2020. GetOrganelle: A fast and versatile toolkit for accurate de novo assembly of organelle



- genomes. *Genome Biol.* 21:241. <https://github.com/Kinggerm/GetOrganelle>
- Katoh K, Standley DM. 2013. MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability. *Mol Biol Evol.* 30:772–780.  
<https://doi.org/10.1093/molbev/mst010>
- Kolmogorov M, Yuan J, Lin Y, Pevzner PA. 2019. Assembly of long, error-prone reads using repeat graphs. *Nat Biotechnol.* 37:540–546.  
<https://www.nature.com/articles/s41587-019-0072-8>
- Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH, Phillippy AM. 2017. Canu: Scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Res.* 27:722–736. <http://genome.cshlp.org/content/27/5/722>
- Li H. 2018. Minimap2: Pairwise alignment for nucleotide sequences. *Method Biochem Anal.* 34:3094–3100. <https://doi.org/10.1093/bioinformatics/bty191>
- Li H, Durbin R. 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics.* 25:1754–1760.  
<https://doi.org/10.1093/bioinformatics/btp324>
- Marçais G, Kingsford C. 2011. A fast, lock-free approach for efficient parallel counting

of occurrences of k-mers. *Method Biochem Anal.* 27:764–770.

<https://doi.org/10.1093/bioinformatics/btr011>

Phillips AL, Ferguson S, Burton RA, Watson-Haigh NS. 2024. CLAW: An automated Snakemake workflow for the assembly of chloroplast genomes from long-read data. *PLOS Comput Biol.* 20:e1011870.

<https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1011870>

Shen W, Le S, Li Y, Hu F. 2016. SeqKit: A cross-platform and ultrafast toolkit for FASTA/Q file manipulation. *PLOS ONE.* 11:e0163962.

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0163962>

Tillich M, Lehwark P, Pellizzer T, Ulbricht-Jones ES, Fischer A, Bock R, Greiner S. 2017. GeSeq versatile and accurate annotation of organelle genomes. *Nucleic Acids Res.* 45:W6–W11. <https://doi.org/10.1093/nar/gkx391>

Wang JR, Holt J, McMillan L, Jones CD. 2018. FMLRC: Hybrid long read error correction using an FM-index. *BMC Bioinf.* 19:1–11.

<https://doi.org/10.1186/s12859-018-2051-3>

Wick RR, Holt KE. 2021. Benchmarking of long-read assemblers for prokaryote whole genome sequencing. *F1000Research.* 8:2138.

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6966772/>

Wick RR, Schultz MB, Zobel J, Holt KE. 2015. Bandage: Interactive visualization of de novo genome assemblies. *Method Biochem Anal.* 31:3350–3352.

<https://doi.org/10.1093/bioinformatics/btv383>

Xian W, Bezrukov I, Bao Z, Vorbrugg S, Gautam A, Weigel D. 2025. TIPPo: A user-friendly tool for de novo assembly of organellar genomes with high-fidelity data.

*Mol Biol Evol.* 42:msae247. <https://doi.org/10.1093/molbev/msae247>

Zhou C, Brown M, Blaxter M, The Darwin Tree of Life Project Consortium, McCarthy SA, Durbin R. 2024. Oatk: a de novo assembly tool for complex plant organelle genomes. *bioRxiv.*:2024.10.23.619857. <https://doi.org/10.1101/2024.10.23.619857>

Zhou W, Armijos CE, Lee C, Lu R, Wang J, Ruhlman TA, Jansen RK, Jones AM, Jones CD. 2023. Plastid Genome Assembly Using Long-read data. *Mol Ecol Resour.* 23:1442–1457. <https://github.com/Bean061/ptgaul>

# Tables

Table 1: Plastid genome assemblies for 23 plant species datasets using subsampled sequencing data with a maximum subsampling rate of 5% (Run Setting A). All datasets were downsampled to 10x genome coverage, and Stage 1 included 10 subsampling steps (N). Depending on the dataset, different maximum sampling rates (P) were used in Stage 1, and different replicate sizes (R) were applied in Stages 2 and 3. NA represents no assemblies in the subsampling-based method and no comparison available.

Species	P	N	R	Length (ptGAUL)	Length (Polap)	Percent identity
<i>Anthoceros agrestis</i>	100	10	5	160010	159938	99.93813
<i>Eucalyptus pauciflora</i>	100	10	5	159841	159945	99.93185

Table 2: Three stages of subsampling-based plastid genome assembly for the *Eucalyptus pauciflora* dataset with Run Setting A. The configuration includes an increasing subsample size up to a maximum subsampling rate of 5%, a step size of 10 in Stage 1, 5 replicates in Stages 2 and 3, and a maximum memory limit of 16 GB. Abbreviations are as follows: iteration in each Stage (I), subsampling rate (Rate) and read-coverage threshold (Alpha); assembly metrics including the number of segments in the assembly (N), the total length of these segments (L), and the number of circular genome paths detected (C); and the draft plastid genome assembly length (Length). Alpha at Stage 3 is the percent identity values between consecutive indices.

Stage	Index	Rate	Alpha	N	L	C	Memory	Time	Length
1	0	0.05	1.00	NA	NA	NA	8	1m	NA
1	1	0.16	0.25	1	118147	0	8	1m	NA
1	2	0.26	0.25	3	130443	4	8	1m	156251
1	3	0.37	0.25	8	253143	8	8	2m	NA
1	4	0.47	0.25	3	130676	4	9	2m	156567
1	5	0.58	0.25	4	173403	4	10	2m	156575
1	6	0.68	1.00	5	132763	4	12	2m	155609
1	7	0.79	1.75	3	130923	4	10	2m	156800
1	8	0.89	1.00	5	272624	4	16	2m	155777
2	0	0.47	0.25	6	265355	4	9	3m	158481
2	1	0.47	0.25	3	132336	4	9	2m	158528
2	2	0.47	0.25	4	180974	4	9	2m	158485
2	3	0.47	0.25	4	193091	4	9	2m	158448
2	4	0.47	0.25	5	247985	4	9	2m	158500
3	0	0.05	NA	NA	NA	NA	.05	0m	159338
3	1	0.29	99.57	NA	NA	NA	.12	.1m	159947
3	2	0.53	100.00	NA	NA	NA	.22	.2m	159944
3	3	0.76	100.00	NA	NA	NA	.31	.3m	159945
3	4	1.00	100.00	NA	NA	NA	.40	.4m	159945

# Figures

Workflow of the subsampling-based plastid genome assembly. The genome assembly procedure is applied repeatedly in Stages 1 and 2.

Figure 1: Workflow of the subsampling-based plastid genome assembly. The genome assembly procedure is applied repeatedly in Stages 1 and 2.

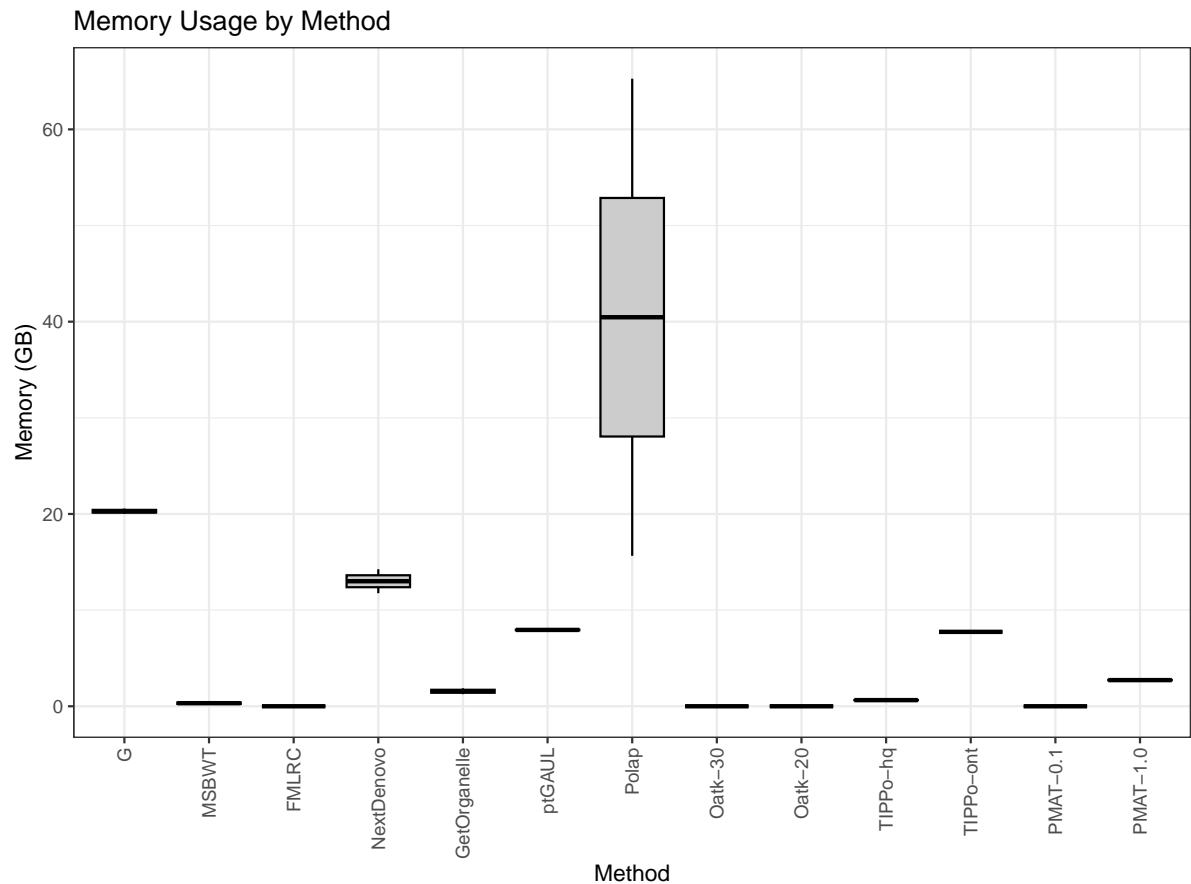


Figure 2: Workflow of the subsampling-based plastid genome assembly. The genome assembly procedure is applied repeatedly in Stages 1 and 2.

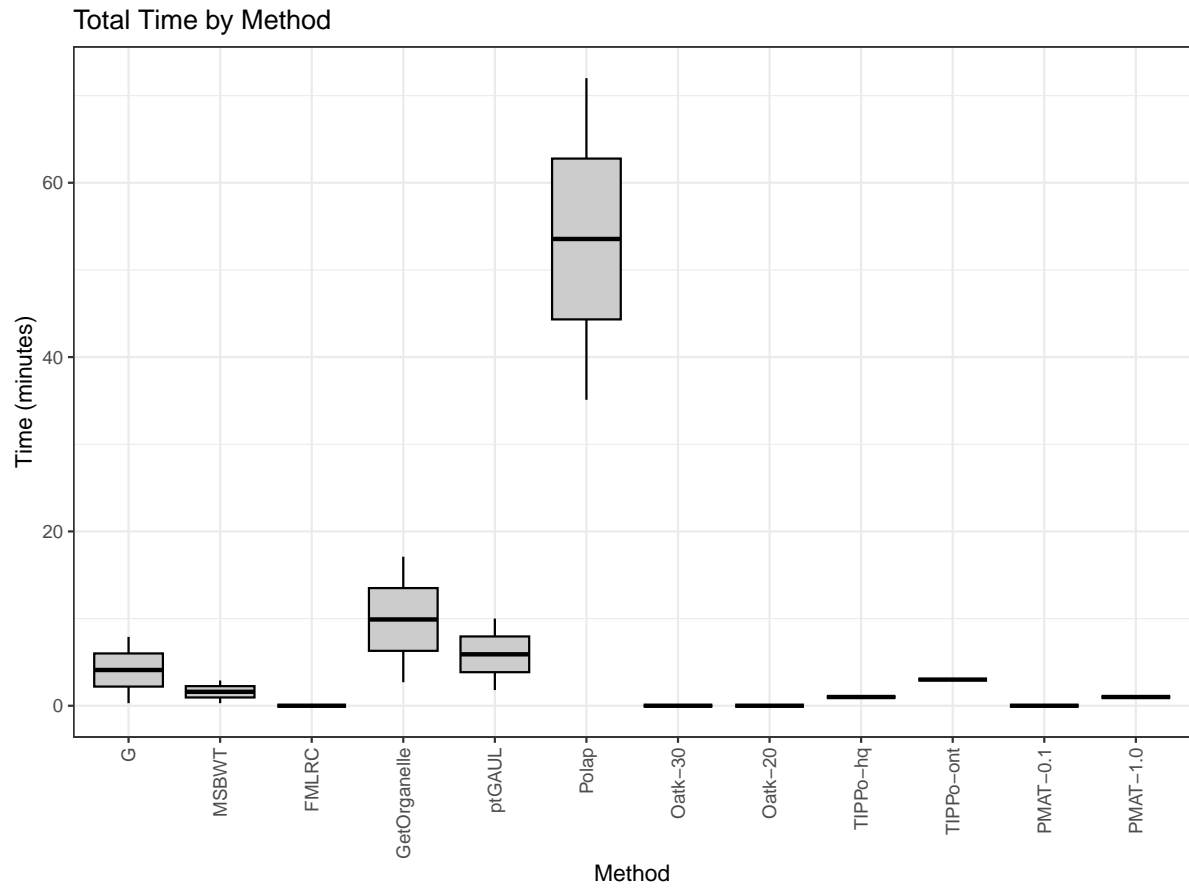


Figure 3: Workflow of the subsampling-based plastid genome assembly. The genome assembly procedure is applied repeatedly in Stages 1 and 2.

## Supplementary Materials

Table S1: Sequencing data for the datasets, including species names and their corresponding taxonomic ranks studied.

Species	Order	Family	Long SRA	Long Size	Long Coverage	Short SRA	Short Size	Short Coverage
<i>Anthoceros agrestis</i>	Anthocerotales	Anthocerotaceae	1	191.5 Mbp	95.58	s	191.8 Mbp	95.70
<i>Eucalyptus pauciflora</i>	Myrtales	Myrtaceae	1	191.5 Mbp	95.58	s	191.8 Mbp	95.70



Table S2: Computer setup for the 23 datasets.

Species		CPU	Cores	Memory	Storage Type
<i>Anthoceros agrestis</i>	E5-2690 v4 @ 2.60GHz		56	251Gi	HDD
<i>Eucalyptus pauciflora</i>	E5-2690 v4 @ 2.60GHz		56	251Gi	HDD

Table S3: Replicate of plastid genome assemblies for 23 plant species datasets using subsampled sequencing data with a maximum subsampling rate of 5% (Run Setting A). All datasets were downsampled to 10x genome coverage, and Stage 1 included 10 subsampling steps (N). Depending on the dataset, different maximum sampling rates (P) were used in Stage 1, and different replicate sizes (R) were applied in Stages 2 and 3.

Species	P	N	R	Length (ptGAUL)	Length (Polap)	Percent identity
<i>Anthoceros agrestis</i>	100	10	5	160010	159936	99.93876
<i>Eucalyptus pauciflora</i>	100	10	5	159841	159936	99.93310

Table S4: Benchmark of `GetOrganelle`, `ptGAUL`, `PMAT`, `TIPPO`, `Oatk` and the method (Run Setting A) presented here in terms of data processing time. NA at the column of `NextDenovo` represents no error-corrected long-read results, resulting in no assemblies in the correction-then-assembly pipelines including `PMAT`, `TIPPO`, and `Oatk`. Abbreviations are as follows: `GetOrganelle` (GO), `ptGAUL` (pG), `NextDenovo` (ND), `PMAT` with `-fc 0.1` (P0.1), `PMAT` with `-fc 1.0` (P1.0), `TIPPO` with `-p onthq` (Thq), `TIPPO` with `-p ont` (Tont), `Oatk` with `-c 30` (O30), and `Oatk` with `-c 20` (O20).

Species	GO	ptG	MSBWT	FMLRC	ND	P0.1	P1.0	Thq	Tont	O30	O20	Polap
<i>Anthoceros agrestis</i>	17.1m	10.0m	2.9m	0m	4.5m	0m	1.0m	1.0m	3.0m	0m	0m	1.2h
<i>Eucalyptus pauciflora</i>	2.7m	1.8m	.3m	0m	.9m	0m	1.0m	1.0m	3.0m	0m	0m	35.1m

Table S5: Benchmark of `GetOrganelle`, `ptGAUL`, `PMAT`, `TIPPo`, `Oatk` and the method (Run Setting A) in terms of peak memory. NA at the column of `NextDenovo` represents no error-corrected long-read results, resulting in no assemblies in the correction-then-assembly pipelines including `PMAT`, `TIPPo`, and `Oatk`. Abbreviations are as follows: `GetOrganelle` (GO), `ptGAUL` (pG), `NextDenovo` (ND), `PMAT` with `-fc 0.1` (P0.1), `PMAT` with `-fc 1.0` (P1.0), `TIPPo` with `-p onthq` (Thq), `TIPPo` with `-p ont` (Tont), `Oatk` with `-c 30` (O30), and `Oatk` with `-c 20` (O20).

Species	GO	ptG	MSBWT	FMLRC	ND	P0.1	P1.0	Thq	Tont	O30	O20	Polap
<i>Anthoceros agrestis</i>	1.87	8.01	0.46	0.00	11.76	0.00	2.75	0.66	7.74	0.00	0.00	65.27
<i>Eucalyptus pauciflora</i>	1.26	7.87	0.18	0.00	14.25	0.00	2.69	0.62	7.74	0.00	0.00	15.65

Table S6: Plastid genome assemblies for 23 plant species datasets using subsampled sequencing data with a maximum subsampling rate of 10% (Run Setting B). All datasets were downsampled to 10x genome coverage, and Stage 1 included 10 subsampling steps (N). Depending on the dataset, different maximum sampling rates (P) were used in Stage 1, and different replicate sizes (R) were applied in Stages 2 and 3. NA represents no assemblies in the subsampling-based method and no comparison available.

Species	P	N	R	Length (ptGAUL)	Length (Polap)	Percent identity
<i>Anthoceros agrestis</i>	100	10	5	160010	159935	99.93689
<i>Eucalyptus pauciflora</i>	99	10	5	159841	159932	99.93560

Table S7: Plastid genome assemblies for 23 plant species datasets using subsampled sequencing data with a maximum subsampling rate of 1%. All datasets were downsampled to 10x genome coverage, and Stage 1 included 10 subsampling steps (N). Depending on the dataset, different maximum sampling rates (P) were used in Stage 1, and different replicate sizes (R) were applied in Stages 2 and 3. NA represents no assemblies in the subsampling-based method and no comparison available.

Species	P	N	R	Length (ptGAUL)	Length (Polap)	Percent identity
<i>Anthoceros agrestis</i>	100	10	5	160010	159935	99.94251
<i>Eucalyptus pauciflora</i>	100	10	5	159841	159935	99.93310

Table S8: Three stages of subsampling-based plastid genome assembly for the *Eucalyptus pauciflora* dataset with Run Setting B. The configuration includes an increasing subsample size up to a maximum subsampling rate of 10%, a step size of 10 in Stage 1, 5 replicates in Stages 2 and 3, and a maximum memory limit of 16 GB. Abbreviations are as follows: iteration in each Stage (I), subsampling rate (Rate) and read-coverage threshold (Alpha); assembly metrics including the number of segments in the assembly (N), the total length of these segments (L), and the number of circular genome paths detected (C); and the draft plastid genome assembly length (Length). Alpha at Stage 3 is the percent identity values between consecutive indices.

Stage	Index	Rate	Alpha	N	L	C	Memory	Time	Length
1	0	0.05	1.00	NA	NA	NA	8	1m	NA
1	1	0.16	0.25	NA	NA	NA	8	1m	NA
1	2	0.26	0.25	1	153400	2	8	1m	153400
1	3	0.36	0.25	3	129973	4	8	1m	155583
1	4	0.47	0.25	3	130068	4	9	1m	156038
1	5	0.57	0.25	3	130536	4	9	2m	156243
1	6	0.68	0.25	3	130917	4	12	2m	156930
1	7	0.78	0.25	5	170704	4	12	2m	156437
1	8	0.89	1.00	4	208162	4	13	3m	156324
1	9	0.99	1.75	3	129602	4	14	2m	155122
2	0	0.57	0.25	4	159399	4	9	2m	158516
2	1	0.57	0.25	3	132295	4	9	2m	158472
2	2	0.57	0.25	2	197574	2	9	2m	158505
2	3	0.57	0.25	4	214682	4	9	2m	158429
2	4	0.57	0.25	3	132256	4	9	2m	158422
3	0	0.05	NA	NA	NA	NA	.08	0m	159223
3	1	0.29	99.48	NA	NA	NA	.12	.1m	159932
3	2	0.52	99.99	NA	NA	NA	.21	.2m	159932
3	3	0.76	100.00	NA	NA	NA	.31	.3m	159932
3	4	0.99	100.00	NA	NA	NA	.40	.4m	159934

Table S9: Three stages of subsampling-based plastid genome assembly for the *Eucalyptus pauciflora* dataset with Run Setting C. The configuration includes an increasing subsample size up to a maximum subsampling rate of 10%, a step size of 50 in Stage 1, 5 replicates in Stages 2 and 3, and a maximum memory limit of 16 GB. Abbreviations are as follows: iteration in each Stage (I), subsampling rate (Rate) and read-coverage threshold (Alpha); assembly metrics including the number of segments in the assembly (N), the total length of these segments (L), and the number of circular genome paths detected (C); and the draft plastid genome assembly length (Length). Alpha at Stage 3 is the percent identity values between consecutive indices.

Stage	Index	Rate	Alpha	N	L	C	Memory	Time	Length
1	0	0.05	1.00	NA	NA	NA	8	1m	NA
1	1	0.10	0.25	4	161519	0	8	1m	NA
1	2	0.15	0.25	10	261384	8	8	1m	NA
1	3	0.20	0.25	1	155145	2	8	1m	155145
1	4	0.25	0.25	3	129647	4	8	1m	155380
1	5	0.30	0.25	3	130705	4	8	1m	156598
1	6	0.35	0.25	4	156123	4	8	1m	156499
1	7	0.40	0.25	3	130637	4	8	1m	156583
1	8	0.45	0.25	3	130116	4	9	2m	155977
1	9	0.50	0.25	4	177242	4	9	1m	156759
1	10	0.55	0.25	6	221747	4	9	2m	155102
1	11	0.60	0.25	4	154713	4	9	2m	154635
1	12	0.65	0.25	4	166791	4	9	2m	155916
1	13	0.70	0.25	6	184437	8	10	2m	NA
1	14	0.75	0.25	4	274222	4	12	3m	156546
1	15	0.80	1.00	4	209428	4	11	3m	155866
1	16	0.85	1.75	3	131009	4	10	2m	156874
1	17	0.90	1.00	5	259272	4	14	2m	156459
1	18	0.95	1.75	3	129146	4	13	3m	153495
1	19	1.00	2.50	3	130928	4	10	2m	156854
2	0	0.35	0.25	3	132312	4	8	2m	158446
2	1	0.35	0.25	4	211664	4	8	2m	158546
2	2	0.35	0.25	3	132387	4	8	2m	158585
2	3	0.35	0.25	1	158527	2	8	2m	158527
2	4	0.35	0.25	3	132021	4	8	2m	158176
3	0	0.05	NA	NA	NA	NA	.05	0m	159377
3	1	0.29	99.58	NA	NA	NA	.12	.1m	159942
3	2	0.53	100.00	NA	NA	NA	.22	.2m	159938
3	3	0.76	100.00	NA	NA	NA	.31	.3m	159938
3	4	1.00	100.00	NA	NA	NA	.40	.4m	159939



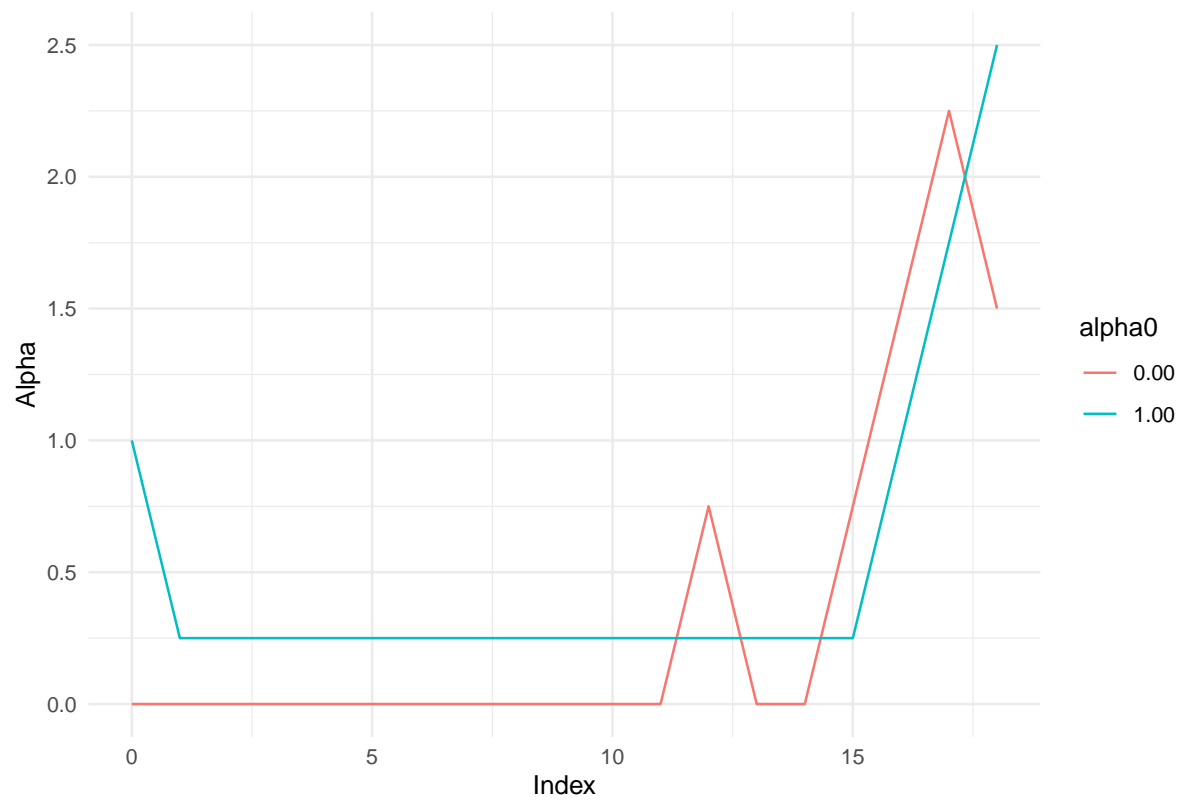


Figure S1: Line plot of read-coverage thresholds versus subsample size index in Stage 1 of the subsampling-based assemblies for *Eucalyptus pauciflora*.

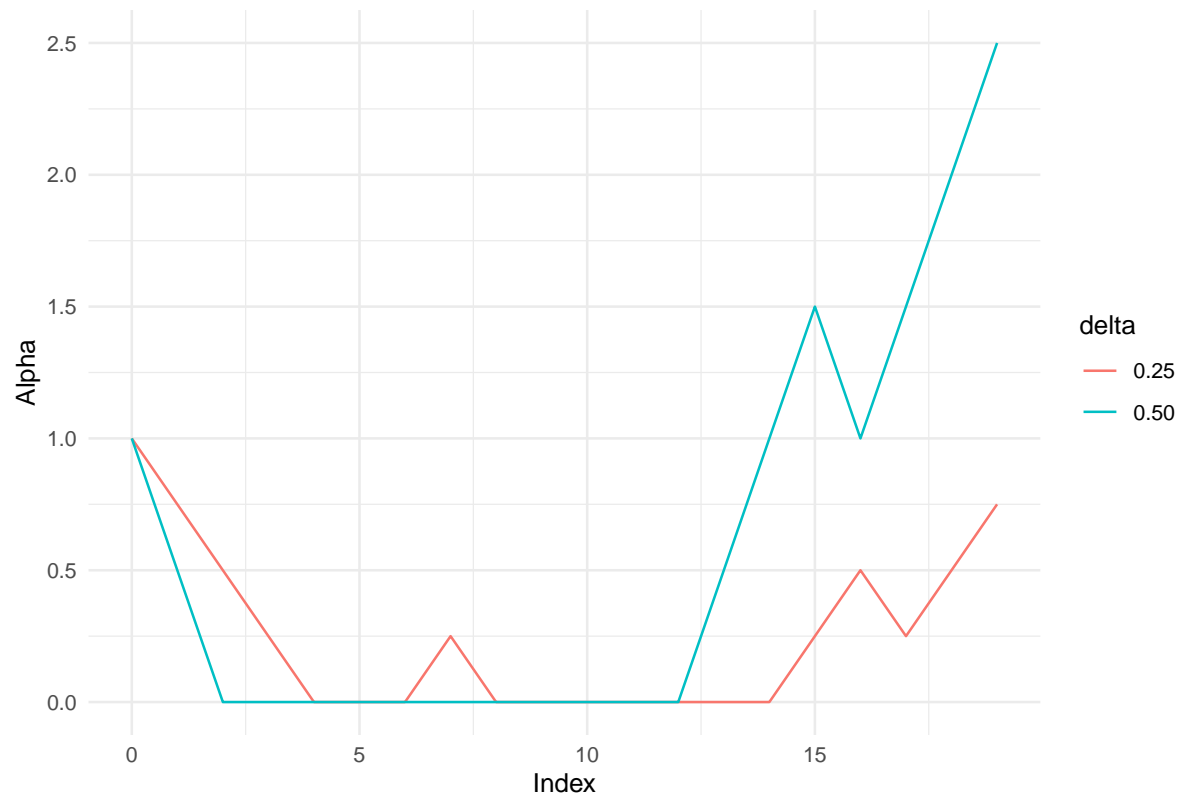


Figure S2: Line plot of increment size versus subsample size index in Stage 1 of the subsampling-based assemblies for *Eucalyptus pauciflora*.

# Code

Polap (Plant Organelle Long-read Assembly Pipeline v0.4.3.7) is available at <http://github.com/goshng/polap>. A quick start guide of the subsampling-based plastome assembly is provided for use on a Linux system with an Internet connection. A detailed guide is also available, offering a step-by-step explanation of test of the procedures outlined in the quick start.

## Requirements

- **Operating System:** Linux (not compatible with macOS or Windows)
- **Dependencies:** Requires [Bash](#) ( $\geq 5.0$ ) and [Miniconda](#)

## Quick Start

To replicate the results presented in this manuscript on a Linux computer with `git` installed and an Internet connection, follow the steps below. Most steps complete in a relatively short time, except for the final step, which includes both data downloading and full analysis:

```
mkdir -p all/polap/cflye1
```

```
cd all/polap/cflye1
```

```
git clone https://github.com/goshng/polap.git
```

```
bash polap/src/polap-data-cflye -y install conda
```

Log out and back in to the terminal.

```
cd all/polap/cflye1
```

```
source ~/miniconda3/bin/activate
```

```
bash polap/src/polap-data-cflye setup conda
```

```
bash polap/src/polap-data-cflye -y install minimal
```

```
bash polap/src/polap-data-cflye setup polap
```

Log out and back in to the terminal.

```
conda activate polap
```

```
polap-data-cflye delete-polap-github
```

```
polap-data-cflye sample-csv polap-data-v2.csv test
```

```
polap-data-cflye -y download-test-data
```

```
# run time: about 1 hour
```

```
polap-data-cflye local-batch Taxon_genus t off
```

```
polap-data-cflye -y install-getorganelle
```

```
# polap-data-cflye -y download-pmat
```

```
# polap-data-cflye -y install-pmat
```

```
polap-data-cflye sample-csv polap-data-v2.csv all on  
# edit the CSV file if necessary  
polap-data-cflye local-batch each
```

Now, go to step 10 of the next subsection to create tables and figures.

## Detailed Guide

- 1. Open a new terminal:** Open a new terminal in a Linux computer, such as one with Ubuntu.
- 2. Install Miniconda:** Download and install [Miniconda](#) using the [instructions](#). The following is a script that works at the time of writing this manuscript. Otherwise, one could easily find a resource for the installation.

```
mkdir -p ~/miniconda3  
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh \  
-O ~/miniconda3/miniconda.sh  
bash ~/miniconda3/miniconda.sh -b -u -p ~/miniconda3  
rm ~/miniconda3/miniconda.sh
```

After installing, close and reopen your terminal application.

**3. Setup the conda channels:** If you did not close and reopen a new terminal, please do so. Then, execute the followings to setup the conda channels for polap.

```
source ~/miniconda3/bin/activate  
  
conda config --add channels bioconda  
  
conda config --add channels conda-forge  
  
conda config --set channel_priority strict
```

**4. Install the Bioconda Polap package:** You setup polap and polap-fmlrc conda environments using [Polap](#) conda package.

```
conda create -y --name polap polap=0.4.3.7.6
```

**5. Installation of Flye for disjointig filtering:** Note that Flye with disjointig filtering feature is a slightly modified version of the original Flye. You activate the polap conda environment and setup polap-fmlrc environment

```
conda activate polap  
  
conda install -y goshng::cflye  
  
base_dir=$(dirname "$(command -v polap)") && \  
  
conda env create -f $base_dir/polap-conda-environment-fmlrc.yaml
```

**6. Polap assemble run with a test dataset:** This tests the basic execution of the polap command.

```
wget -q https://github.com/goshng/polap/archive/refs/tags/0.4.3.7.6.zip
unzip -o -q 0.4.3.7.6.zip
cd polap-0.4.3.7.6/test
polap assemble --test
```

**7. Plastid genome assembly with *Eucalyptus pauciflora* dataset:** Your assembled plastid genome sequence will be o/ptdna.0.fa.

```
polap x-ncbi-fetch-sra --sra SRR7153095
polap x-ncbi-fetch-sra --sra SRR7161123
polap disassemble -l SRR7153095.fastq \
    -a SRR7161123_1.fastq \
    -b SRR7161123_2.fastq
```

**8. Check the accuracy of the plastid genome assembly:** We use the Polap disassemble command with *Eucalyptus pauciflora* dataset and check its similarity with its known plastid genome sequence Your assembled plastid genome sequence will be o/ptdna.ref.0.fa. The text file named o/0/mafft/pident.txt has the percent identity between the assembled ptDNA and the known reference.

```

polap get-mtdna --plastid --species "Eucalyptus pauciflora"

cp o/00-biopproject/2-mtdna.fasta o/ptdna-reference.fa

polap disassemble \

    --disassemble-i 1 \

    --stages-include 3 \

    -l SRR7153095.fastq \

    -a SRR7161123_1.fastq \

    -b SRR7161123_2.fastq \

    --disassemble-align-reference \

    --disassemble-c o/ptdna-reference.fa


mkdir -p o/0/mafft

polap mafft-mtdna \

    -a o/ptdna-reference.fa \

    -b o/0/disassemble/2/pt.subsample-polishing.reference.aligned.1.fa \

    -o o/0/mafft

cat o/0/mafft/pident.txt

```

## 9. Batch script that creates the results in the manuscript:

```

polap-data-cflye -y install-getorganelle

# polap-data-cflye -y download-pmat

```



```
# polap-data-cflye -y install-pmat
```

```
polap-data-cflye example-data polap-data-v2.csv all on
```

```
polap-data-cflye local-batch each
```

**10. Tables in the manuscript:** Tables in Markdown format will be generated and saved in the `man` directory after executing the following command. You should download a precompiled binary version 0.8.1 of **Bandage** genome assembly graph visualization tool from [the official Bandage GitHub](#).

```
polap-data-cflye -y install-bandage
```

```
# Install xelatex if necessary ...
```

```
# sudo apt-get install texlive texlive-latex-recommended texlive-xetex
```

```
# sudo apt-get install texlive-fonts-recommended texlive-fonts-extra texlive-lang-all
```

```
polap-data-cflye -y install-man
```

```
polap-data-cflye -y download-man
```

```
polap-batch-v2.sh
```

```
polap-data-cflye -y make-man
```