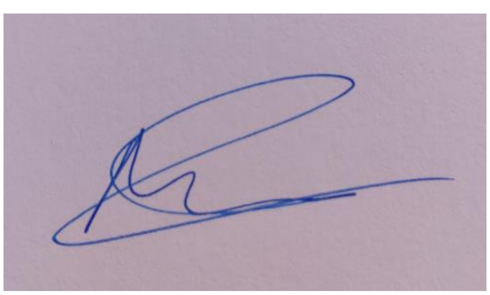
A gene co-expression network reveals coordinated rhythmic gene expression patterns in the charophyte *Klebsormidium nitens*

Máster en Genética y Biotecnología. Universidad de Sevilla.

Trabajo Fin de Máster. 17 de Septiembre de 2020.



Firmado por ROMERO CAMPERO FRANCISCO JOSE - 48863139F el día 02/09/2020 con un certificado emitido por AC FNMT Usuarios

Alumno: Marcos Elizalde Horcada

Director: Francisco José Romero-Campero

A gene co-expression network reveals coordinated rhythmic gene expression patterns in the charophyte *Klebsormidium nitens*

Marcos Elizalde-Horcada[1,*], Francisco J. Romero-Campero[1, 2,*]

[1]Institute for Plant Biochemistry and Photosynthesis, University of Seville – CSIC, Américo Vespucio 49, 41092, Seville (Spain)

[2]Deparment of Computer Science and Artificial Intelligence, University of Seville, Reina Mercedes s/n, 41012, Seville (Spain)

***Correspondence:**
Corresponding Authors
marcos.elizaldeh@gmail.com, fran@us.es

Keywords: RNA-seq, Transcriptomics Diurnal rhythmic genes, Gene network analysis, Klebsormidium, Charophytes,

Abstract

Klebsormidium nitens is an emerging model organism within charophytes for the study of the adaptation to terrestrial environments, being considered a link between algae and land plants. In this study we re-analyze some previously published data from a different perspective, using up to date software tools to determine which part of diurnal rhythmic expression of *Klebsormidium* genome exhibits rhythmic diurnal expression patterns. We found that 62.19% of *Klebsormidium* genes are expressed rhythmically during 24 hours periods alternating 12 hours of light and 12 hours of dark. This result provides more evidence of the key place occupied by this model organism between microalgae that typically present a percentage of rhythmic genes greater than 80% and land plants whose rhythmically expressed genes do not normally exceed 30% of their genomes. The construction and analysis of a gene co-expression network reveals some new insights of this genome and how it might be structured.

1 Introduction

Klebsormidium nitens (*K. nitens*) is a charophytic algae consisting in multicellular and non-branching filaments with a lack of specialized cells. Having tolerance to drought and freezing, it is a species adapted to land but can also growth in fresh water. This makes *K. nitens* a key species in the study of plant terrestralization (Hori et al., 2014). *K. nitens* has emerged as an interesting model algae in the context of adaptation to terrestrial environments, since it has acquired during evolution some genes that we found today to be specific to land plants. It possesses gene groups commonly found in other plant species and lacking in microalgae genomes, suggesting some proteins resemble those of land plants more than those of chlorophyta microalgae analyzed (Ferrari et al., 2019). Also, it has been shown that this organism produces some plant hormones and homologues of some of the signaling intermediates required for hormone actions in vascular land plants, as well as a primitive system for high light intensity damage protection. Being an organism between unicellular microalgae and multicellular land plants, it becomes a natural candidate to study the mechanisms and adaptations that led to terrestrial colonization, including diurnal rhythmic expression patterns. In this study we reanalyze *K. nitens* previously published RNA-seq data (Ferrari et al., 2019) with two different goals. First, we aim at identifying genes with rhythmic expression patterns using an up to date non parametric method called RAIN (Rhythmicity Analysis Incorporating Non-parametric Methods). Second, we characterized co-expression gene patterns by constructing and analyzing a gene co-expression network.

The process to determine and classify rhythms in gene expression has been an active research field developing software tools that can deal with the huge amount of experimental and biological noise produced by omics techniques. We try to address this problem because it has been suggested that the establishment of multicellularity rather than land colonization decreased the diurnal regulation of gene expression. In this respect, *K. nitens* can be considered a representative of the link between algae and land plants. We seek to determine if *K. nitens* shows a decrease of rhythmicity compared to microalgae or not. Furthermore, we constructed a co-expression network of these diurnal rhythmic genes which has not been done yet, hoping the study of the topology and structure of the network would reveal some new insights of *K. nitens* biology.

2 Materials and methods

2.1 Materials

- Data (RNA-seq reads, reference genome and annotation)
- HISAT2 software (https://daehwankimlab.github.io/hisat2/)
- StringTie software (https://ccb.jhu.edu/software/stringtie/)
- SAMtools (http://www.htslib.org/)
- MobaXterm (https://mobaxterm.mobatek.net/)
- R (https://cran.r-project.org/, version 4.0.2)
- RStudio (https://rstudio.com/, version 1.2.5042)
- Bioconductor packages: ballgown, genefilter, clusterProfiler, rain, ggplot2, dplyr, tidyr, annafy, pathview, WCGNA, cluster)
- Cytoscape 3.8.0 (https://cytoscape.org/)
- Other packages: Hisat2, StringTie, VennDiagram, robustbase, Factominer, factoextra, igraph

The reference genome and the annotation were downloaded from *Klebsormidium nitens* NIES-2285 genome project online platform (http://www.plantmorphogenesis.bio.titech.ac.jp/~algae_genome_project/klebsormidium/)

2.2 Sample Processing

The first step of the analysis consisted on the generation of the workspace containing the appropriate directories and files.

The sample processing performed in this study is based on various software tools called the “new Tuxedo package”, which include HISAT, StringTie and Ballgown as described in Science Protocols (Pertea et al., 2016). Figure 1 shows these and the main steps performed by each tool. Due to the large number of samples to process, 39 samples in overall, and the time needed to process each one, approximately 2 hours, we developed a fully automatic bash pipeline. This also prevented the accumulation of code bugs in the error prone process of carrying out each step manually. The steps of the sample processing are described below, whereas the automatic workflow will be explained in the bash-scripting section.

Creation of genome index

The first step was to build a HISAT2 index from both the reference genome annotation files with the hisat2-build function. This function uses a data structure based on the Burrows-Wheeler transform through the blackwise algorithm of Karkkainen (Burrows and Wheeler, 1994). This allows not only for data compressing, but also for reversibility. To generate the index itself, we first need to extract the splice sites and exons information from the annotation file, with Python scripts provided by the HISAT2 package (Kim et al., 2015). With this information as arguments for the hisat2-build function a set of 6 files constituting the index is created. This is all we need for further aligning reads to this reference. Code is shown in BOX 1.

BOX 1. Creation of genome index

Note that genome and annotation files must be stored in the genome and annotation folders. With annotation as our current directory, we run the code as it follows:

```
extract_splice_sites.py annotation.gtf > annot_splice.ss

extract_exons.py annotation.gtf > annot_exons.exon

hisat2-build --ss annot_splice.ss --exon annot_exons.exon ../genome/genome.fa index
```

Sample downloading and quality control

Each sample was downloaded using the command fastq-dump with the SRA accession number as an argument. The sequenced reads are stored in fastq format. Each read in a fastq file starts with a sequence identifier preceded by the symbol @, the read sequence, a separator, which is simply a plus (+) sign and the base quality scores. We passed these files to the fastqc (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/) function in order to ensure the data has enough quality for further steps. This is a Java-based quality control tool for high throughput sequencing data which provides a modular set of analyses which gives us a quick impression of whether our data has problems of which we should be aware before further analysis.

Align the RNA-seq reads to the genome

The protocol begins by mapping reads from each sample against a reference genome to identify their genomic positions. HISAT2 is an alignment software that reads fastq files and assign the sequence reads to a position with respect to a known reference genome. This mapping information allows us to generate subsets of the reads corresponding to each gene. In this step, HISAT2 uses the Burrow-Wheeler transform as when creating the index, allowing rapid mapping even running on a conventional desktop.

Sort and convert the SAM files to BAM

The HISAT2 alignment produces a SAM file for each sample, which is a text format for storing sequence data describing mapping information. These files typically contain a header with information about the alignment and a number of lines representing each single read with eleven mandatory tab separated fields of information (Li et al., 2009). The SAM to BAM converting through SAMtools is just a matter of saving disk space. This process generates a BAM file for each SAM file which contains the same information of alignment we had in the SAM file but binary encoded.

Assemble and quantify expressed genes and transcripts

Transcript assembly and quantification can be effectively achieved with StringTie (Pertea et al., 2015). StringTie uses a genome-guided transcriptome assembly approach along with concepts from de novo genome assembly to improve transcript assembly. It uses the optimization technique of maximum flow in a specially constructed flow network to determine gene expression levels, and does so while simultaneously assembling each isoform of a gene. It then removes the reads associated with that transcript and repeats the process, assembling more isoforms until all the reads are used, or else until the number of reads remaining is below the user-adjustable level of transcriptional noise. Even though StringTie does not need the reference genome for transcript assembly, we provided *K. nitens* annotation to facilitate the process. This can be helpful while reconstructing low-abundance genes for which the number of reads is too low.

Merge transcripts from all samples

The genes and isoforms present in one sample are usually different to those present in all other samples. And also coverage might be different for each exon, or some parts could be missing. Merging all assemblies with StringTie's merge function solves this problem for us. This function can find consistency between assemblies and reference annotation, but also between samples which are consistent with each other, being able to automatically detect new genes and new isoforms whether or not they appear in standard annotation.

Comparing transcripts with the reference annotation

This step is optional and allows us to compare the percentage of similarity between StringTie's merge and the reference annotation. The output, gffcmp.stats file contains information and statistics for different gene features, such as merge-annotation overlapping, proportion of genes from annotation correctly reconstructed, total number of novel exons, and so on. For our particular analysis, knowing both the merge and the annotation file are highly similar is enough. All of the steps above are shown in BOX 2.

BOX 2. Sample processing

Download each sample

```
fastq-dump  accesion_number
```

Run quality analysis

```
fastqc  sample_n.fastq
```

Mapping reads against reference genome

```
hisat2 –dta – x ../../genome/index -U sample_n.fastq –S sample_n.sam
```

Sort and convert SAM file into BAM file

```
samtools sort –o sample_n.bam sample_n.sam
```

Assemble and quantify gene expression

```
stringtie –G annotation.gtf –o sample_n.gtf –l sample_n  sample_n.bam
```

Transcriptome merging

```
stringtie –merge –G input_gtf –o output_gtf mergelist.txt
```

Bash-scripting

The whole protocol explained above can be made for each sample, handwriting the code each time. This can be an educational but error prone process, and it can be worth doing with a low number of samples (i.e. 4-8 samples). In this experiment, we started with 39 samples, so we took some time to learn the basics of bash-scripting in order to achieve work automation on sample processing. For this purpose, we started creating a script called sample_processing.sh, specifying three variables: sample folder, accession number and sample number, plus the whole sample processing code described above, filling the appropriate gaps with this three variables in order to be able to loop through the 39 samples. To get this done, we created another file called knitens.params where we stored manually all the information needed to be coded in each variable for each loop cycle: the folder where the samples are going to be downloaded and processed, the number of samples and each of the 39 individual accession numbers. Finally, we made another script called knitens.sh, which works as a parameter of the sample_processing.sh script. We could call this one the executor script, because his job is to read each parameter in knitens.params, generate each 39 sample folders and launch 39 times the sample_processing.sh script with each parameter, completing the sample processing of all the samples in approximately 7 hours. These scripts are available on GitHub (https://github.com/marcos-bioinformatics) and are briefly shown in Figure 2.

2.3 Analysis of the processed data

Experimental design

Once we have all the samples processed, this is, mapped to the reference genome, merged and quantified, we can analyze the data in multiple ways. First we set the experimental design and loaded it into R. 39 samples collected every two hours from ZT1 to ZT25 in triplicates, with light/darkness condition associated to each sample. After this, we read every genetic expression data from each processed sample and construct a gene expression matrix with the ballgown and gexpr functions from the Ballgown package (Fu et al., 2020). After labeling appropriately each column name, we constructed a mean expression matrix containing the mean expression of the 3 replica per zeitgeber time.

Principal Component Analysis (PCA)

With the goal of summarizing and visualize the data, we first performed a Principal Component Analysis (PCA), expressing this observations as a set of a few new variables called principal components, which correspond to a linear combination of the originals. PCA allows us to identify the principal components along which the variation of the data is maximal as well as visualize this graphically. We were particularly interested in discovering a circadian patron while representing the two principal components. The function PCA from the FactoMiner package (Lê et al., 2008) takes the transpose gene expression matrix as an input, which output is a list of objects containing information about the analysis. We were mostly interested in the eigenvalues, which measure the amount of variation retained by each principal component and visualizing them. For this purpose, we extracted and visualized the results of PCA using the factoextra package (https://CRAN.R-project.org/package=factoextra). It is important to note here that we did perform two PCA analysis in our study, one with the gene expression matrix, and one with a subset of this matrix containing just those genes considered circadian by the RAIN package with a p-value of 0.01 or lower, as explained below.

RAIN

After the PCA analysis, we wanted to determine how many of the sample's genes showed some kind of diurnal rhythmic expression patron. This is a really difficult problem to approach, since classic methods based on Fourier theory are often hampered by the complex and unpredictable characteristics of experimental and biological noise. We selected the RAIN package (Thaben and Westermark, 2014) to address this problem, consisting in a nonparametric method for detection of rhythms of pre-specified periods in biological data (zeitgeber time in our study), particularly different wave forms. This package, built from another nonparametric program, JTK_CYCLE (Hughes et al., 2010), has some improvements. Over parametric methods, RAIN doesn't assume the noise variance is both Gaussian and independent of measurement magnitude. And over other nonparametric methods such as JTK_CYCLE, RAIN does not assume wave forms to be perfectly symmetric, not comparing the rising and falling parts of the wave and testing them independently. Also, RAIN package uses Benjamini-Hochberg (Benjamini and Hochberg, 1995) correction for multiple testing due to varying umbrella peaks and phases. This method has shown a good level of detection power while keeping the false discovery rate low, and has been validated against independent data. The rain function takes the gene expression transpose matrix, and the following arguments have to be specified: time difference between two data points (2h), the period to test for (24h) and the number of series for each sample (3). The output consists in a table in which the p-value, phase, peak.shape and period are shown. We took all the genes with a p-value lower than 0.01.

Hierarchical Clustering

We performed agglomerative hierarchical clustering to assess the similarity and grouping of the samples. This was done by the HCPC function from the FactoMineR package which takes the PCA object from the PCA function as input, building a hierarchy tree. This function uses flexible UPGMA cluster analysis, based on the Lance and Williams clustering strategy (Lance and Williams, 1967).

We ran this analysis twice, once with the PCA results but also with the PCA results subsetting those genes considered circadian by the RAIN package, hoping the subset would reduce the noise achieving a better grouping of the data.

Rain Clustering

As we aimed to identify not only which genes showed some kind of diurnal rhythmic expression patron but also at what zeitgeber times were maximally and minimally expressed, we looped the whole matrix, reading and determining the highest point and the lowest point of expression for each gene, and the time correspondence for both. This allowed us to catalog these diurnal rhythmic genes into groups for each zeitgeber time, being able to conduct further studies such as per-ZT gene ontology analysis.

Gene Ontology Analysis

The next step was performing a Gene Ontology analysis, retrieving functional information about our subset of circadian genes to try to understand the underlying biological processes. This was achieved using the clusterProfiler package (Yu et al., 2012) and the appropriate annotation for Klebsormidium nitens, org.Knitens.eg.db. We first set the background universe. After that, we created a list of genes for each zeitgeber time (using rain package's peak criteria) and saved them onto text files. This would allow us to process each set of genes through the enrichGO function separately, obtaining an ordered output of all of the zeitgeber time's GO enriched terms. The proper arguments were manually provided by us for the enrichGO function: biological process, Benjamini-Hochberg for multiple testing adjust method, and a p-value cutoff equal to 0.05. With this setup, we ran the function to each list of genes, outputting the same number of Go enriched terms lists and different plots.

KEGG Analysis

Complementary to the GO enrichment, the clusterProfiler provides an enrichKEGG function to construct a KEGG pathway mapping. For this purpose, we took a subset of the background universe containing KO terms, as our model organism does not have the appropriate term annotation, so this process had to be done manually. Prior to subsetting and eliminating not assigned values, we ran the enrichKEGG function for each cluster and zeitgeber time gene lists with a qvalueCutoff equal to 0.05, generating a data frame with the results. After this, we generated graphical representations of these enriched pathways with the pathway function from the pathway package (Luo et al., 2013).

Network construction and visualization

To know more about these diurnal rhythmic genes and their relationships, we constructed a gene network using a correlation matrix based on Pearson's correlation as an input for the igraph package (Csardi and Nepusz, 2006). Firstly, we assessed different threshold values for correlation and R squared as a measure of adjustment to the scale free property. Once this was done, we created the network with the appropriate cutoff value (Correlation threshold = 0,975). We performed different analysis on the network prior to visualizing it such as power-law fitting, degree distribution, network clustering coefficient, average path length and hub distribution. We also performed a Montecarlo method generating 1000 random networks to evaluate the small world property of our gene network. Further to this, we performed two types of clustering to the network: agglomerative clustering with R base hclust function and PAM clustering with the pam function from the cluster package (Maechler et al., 2019). GO and KEGG analysis were performed for both this clusters, as well as for some interesting genes showing high hub scores, high transitivity values or those who were transcriptional factors. The network was loaded onto Cytoscape for visualization (Shannon et al., 2003).

3 Results

3.1 Sample processing

There were not any reported issues while reviewing the samples with the fastqc package, all of them showing high per base sequence quality scores, high per sequence quality scores and an appropriate per sequence GC content. The executor script knitens.sh was launched and the whole sample processing shown in Figure X was correctly performed for each of the 39 samples. For our particular analysis, knowing that StringTie's merge file and the annotation file were highly similar is enough. The number of reads per sample and the alignment percentage is shown in Table 1. It is important to note that samples ERR2820841 and ERR2820842 showed the same number of reads and the same percentage of alignment. This might be an error uploading the samples, as they seem to be the same file.

3.2 Principal Component Analysis (PCA), Rain and Hierarchical Clustering

Because we were particularly interested in discovering a circadian gene expression patron while representing the two principal components and cluster creation, both the PCA and the hierarchical clustering were performed before and after applying the rain package to the data. The rain package classified 10754 out of 17290 genes (62.19%, p value < 0.01) showing some consistent diurnal rhythmic expression pattern over the day, which we considered diurnal rhythmic genes for *Klebsormidium nitens*. This was pretty surprising, since this same data was analyzed in the original study claiming just 39.4% (p value < 0.05) of *K. nitens* genes to be circadian. This was probably due to the use of the JTK_Cycle algorithm which seems to detect less rhythmic genes than the rain packed used in our study. So far, this is consistent with algae tending to be on average more rhythmic than multicellular land plants, thus having a lot of processes whose gene expression is regulated by alternating day and night intervals. But it does not seem as pointed out in Nanyang's study that *K. nitens* has a decreasing number of rhythmic genes compared with other algae from Archaeplastidia. Our data analysis suggests that *K. nitens* still has a strong diurnal gene expression control. Again, this might be related to the use of a different software for determine circadian patrons.

As a result of applying the rain package, the percentage of variance explained by the two principal components grew from 54.7% to 63.8%, eliminating some noise from the original data. On the other hand, the hierarchical clustering did not show any differences before and after applying the rain package (see Figure 3). We found 3 clusters which correspond to dawn, day and night. The dawn cluster, shown in blue was the most consistent one, grouping up ZT1 and ZT25 samples together, as these samples were taken at the same time of the day. The day cluster, grey colored, had some inconsistencies, as it did not match the triplicates together, probably because they had not been taken rigorously over time. The night cluster had the same issues and could not match the triplicates together. Despite this, hierarchical clustering revealed a diurnal rhythmic expression pattern and the samples were consistently separated in three distinct groups: dawn (ZT1 and ZT25), day (ZT3 to ZT13) and night (ZT15 to ZT23).

3.3 Gene Ontology analysis (GO) and KEGG analysis

We performed a gene ontology analysis with clusterProfiler on the three clusters and for each zeitgeber time separately. No terms were found for ZT3, ZT5 and ZT23 individually. KEGG pathway mapping was performed for each zeitgeber time, not being able to find any mapping for ZT1, ZT13, ZT15, ZT19 and ZT23. Figure 3 shows the most significant GO and KEGG terms found for each cluster and zeitgeber time. Supplementary Table 1 shows the most representative GO and KEGG terms and genes associated for each cluster and zeitgeber time.

<u>Dawn</u>

This cluster, comprising ZT1 and ZT25 showed mostly transmembrane transport as well as lipid and organic acid biosynthesis/metabolism, probably as a response to dawn light, preparing the algae for the sunlight. Looking up closely for ZT1 and ZT25 individually, clusterProfiler did not find any significant terms regarding photosynthesis, which were low or even absent in mostly all of zeitgeber times. It could be possible that genes regulating this process are constitutive and do not show a significant pattern over the day. Pathway mapping was found for ZT25 showing glycine, serine and threonine metabolism and carbon fixation.

<u>Day</u>

Most of the terms found in this cluster were amino-acid metabolism related, indicating a high level of RNA traduction. No photosynthesis related terms were found, but KEGG mapping showed several genes related to oxidative phosphorylation, especially regarding NADH dehydrogenase, and F-type ATPase at ZT3, which is consistent with the sunlight before dawn. Amino acid metabolism, RNA metabolism and DNA replication appear to be mapped at the afternoon, from ZT9 to ZT13.

<u>Night</u>

Many protein synthesis related ontology terms were found in this cluster, showing *K. nitens* might be preparing the molecular machinery consisting of protein for the biological processes that need to be performed during the coming 12 hours light period.

3.4 Network construction and analysis

The co-expression network based on Pearson's correlation was successfully created with a cutoff value equal to 0.975. This cutoff value showed the best scale free model fit (R^2). The node degree distribution was tested against a power law and the p-value of the Kolmogorov-Smirnov test was 0.99987 as shown in Figure 4. After determining this, we wanted to assess the network's fit to a small world network. For this purpose, we calculated the network clustering coefficient (0.4111) and the average path length (11.961) and compared this using a Montecarlo method simulating 1000 networks with the same number of edges and nodes. Not a single simulated network had a higher clustering coefficient than our network, but the average path length for the simulated networks was more than 3 fold lower than our network, determining we cannot say our network is a small world one. In summary, our network, comprising 10754 nodes and 67135 edges was a free-scale but not a small world network. After loading and removing some unconnected elements, which did not seem to have correlation on Cytoscape, the resulting network comprised 5731 nodes and 66776 edges, showing some circadian distribution.

After this, an agglomerative cluster was performed using both PAM and Hclust methods. We ran both methods for k = 2 to k = 10 and found that the best silhouette fit was for the PAM method taking into account just 2 clusters, with an average silhouette width of 0.55. The silhouette plot for the PAM clustering is shown in Figure 4, and we colored both clusters for better visualization in Figure 5. GO analysis of this clusters revealed cluster number 1 was in charge of amide biosynthetic process, peptide metabolism, RNA processing and translation as well as DNA repairing. Cluster number 2 showed totally different GO terms related to carbohydrate and lipid metabolism, organic acid biosynthesis and transmembrane transport. These clusters are shown in Figure 4. Even though clustering revealed this diferences, we would've liked to discover more consistent clusters finding more subtle GO terms for each of them, being able to understand more deep this co-expression network. This lack of discrimination within the data could be a result of the poor annotation and knowledge regarding the biology and genetics of *K. nitens* to this date.

Prior to the clustering we calculated the hub score and clustering coefficient values for each node, coloring those who had a value above the 95^{th} percentile. The network topology as well as the network hubs and nodes with a high clustering coefficient value are shown in Figure 4. These figures show that the network hubs are located in the middle of the network, and those nodes with high transitivity are more widely distributed. We ran a gene ontology analysis for the highest hub score and clustering coefficient genes and his k=3 nearest neighbors but we did not find any significant GO terms related to this genes. This might be surprising, as these genes play an unquestionable role on the network due to his connectivity characteristics, but it is also unquestionable that *K. nitens* genome lacks a deep understanding against other model organisms and annotation is still on his first steps of development.

As a final step, we managed to identify where the transcription factors were located in the network, coloring them as shown in Figure 5. This kind of distribution recalls the one we saw for high transitivity genes in Figure 4. In fact, most of the transcription factors identified had high transitivity values, which suggests a key role in the organization of the network as they tend to group or cluster genes around them.

The most represented transcriptional factor families were CH3, bZIP, B3 and C2H2. As we can see in Figure 5, the CH3 elements are distributed in the center of the network, while B3, C2H2 and bZIP elements are located on the periphery. This is interesting since we can imagine CH3 elements being genes serving as central connectors while bZIP, C2H2 and B3, which show higher transitivity values have a lot of neighbors who are neighbors between them. This can be consistent with these families being transcriptional activators. We performed a GO analysis on some of this transcriptional factors and his k=3 nearest neighbors and found consistency between the GO terms and known function of these genes in other organisms, mostly RNA processing, DNA processing and chromosome organization, as shown in Table 2.

4 Discussion

This study reveals that *K. nitens* does not show the drastic decrease of genome wide rhythmicity compared to microalgae discussed on the original study. The data reanalysis suggests *K. nitens* has a high value of diurnal rhythmic genes, 62.19% against the 39.4% claimed on the original study (Ferrari et al., 2019). Even though the software to determine rhythmic expression patterns was different, we used a more conservative p-value. This supports our hypothesis that *K. nitens* has an important diurnal gene control, and marks the need of using up to date and experimentally verified tools to address this kind of issues. This result provides more evidence of the key place occupied by this model organism between microalgae that typically present a percentage of rhythmic genes greater than 80% (Monnier et al., 2010; Zones et al., 2015) and land plants whose rhythmically expressed genes do not normally exceed 30% of their genomes (Covington et al., 2008; Michael et al., 2008). The existence of discrepancies while clustering might be explained by a non-rigorous sampling over the zeitgeber times, but nevertheless we were able to see three distinct clusters that suggest there is in fact a circadian clock controlling some of *K. nitens* gene expression. Gene Ontology analysis showed some distinct biological processes for each of the clusters, but did not completely reveal more precise information for each zeitgeber time. We were working with a non-model organism which genome was published just some years ago, so there is a need for further research and proper annotation to be able to discover new insights of *K. nitens* genome and molecular biology. The network construction and analysis revealed some new information about *K. nitens* genome. Some circadian structure can be visualized and important genes such as hubs and high transitivity genes were discovered, some of them well known transcriptional factors related to stress responses (Jakoby et al., 2002; Kiełbowicz-Matuk. 2012). This is a relative new technique which has produced powerful insights from other algae like Chlamydomonas (Romero-Campero et al., 2016). *K. nitens* co-expression network constitutes a modest first step that will require proper annotation and further data collection and analysis, since the GO analysis performed on the network genes did not produce any significant results. Nevertheless, this should generate new hypothesis that can be a start point for future experiments.

5 Conflict of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

6 Author Contributions

M.E.H has developed all the software code and performed all the computational analysis under the supervision of F.J.R.C. M.E.H wrote this manuscript under the supervision of F.J.R.C.

7 Acknowledgments

Special thanks for Francisco Romero-Campero for his guide, teaching and knowledge. Thanks to Alexia Martínez and Sergio Bustamante for their restless support and Valorant matches.

8 Supplementary Material

A supplementary table is available showing GO and KEGG terms for each cluster and zeitgeber time, and a list of genes associated with those terms.

Data and Code Availability Statement

The RNA-seq raw data analyzed in this study is available from the database Gene Expression Omnibus identified with accession numbers ERR2820833, ERR2820834, ERR2820835, ERR2820839, ERR2820840, ERR2820841, ERR2820842, ERR2820843, ERR2820844, ERR2820845, ERR2820846, ERR2820847, ERR2820848, ERR2820849, ERR2820850, ERR2820830, ERR2820831, ERR2820832, ERR2820728, ERR2820729, ERR2820730, ERR2820731, ERR2820732, ERR2820733, ERR2820734, ERR2820735, ERR2820736, ERR2820737, ERR2820738, ERR2820739, ERR2820740, ERR2820741, ERR2820742, ERR2820725, ERR2820726, ERR2820727, ERR2820836, ERR2820837, ERR2820838. The software code developed in this study can be accessed here https://github.com/marcos-bioinformatics.

Figures and Tables

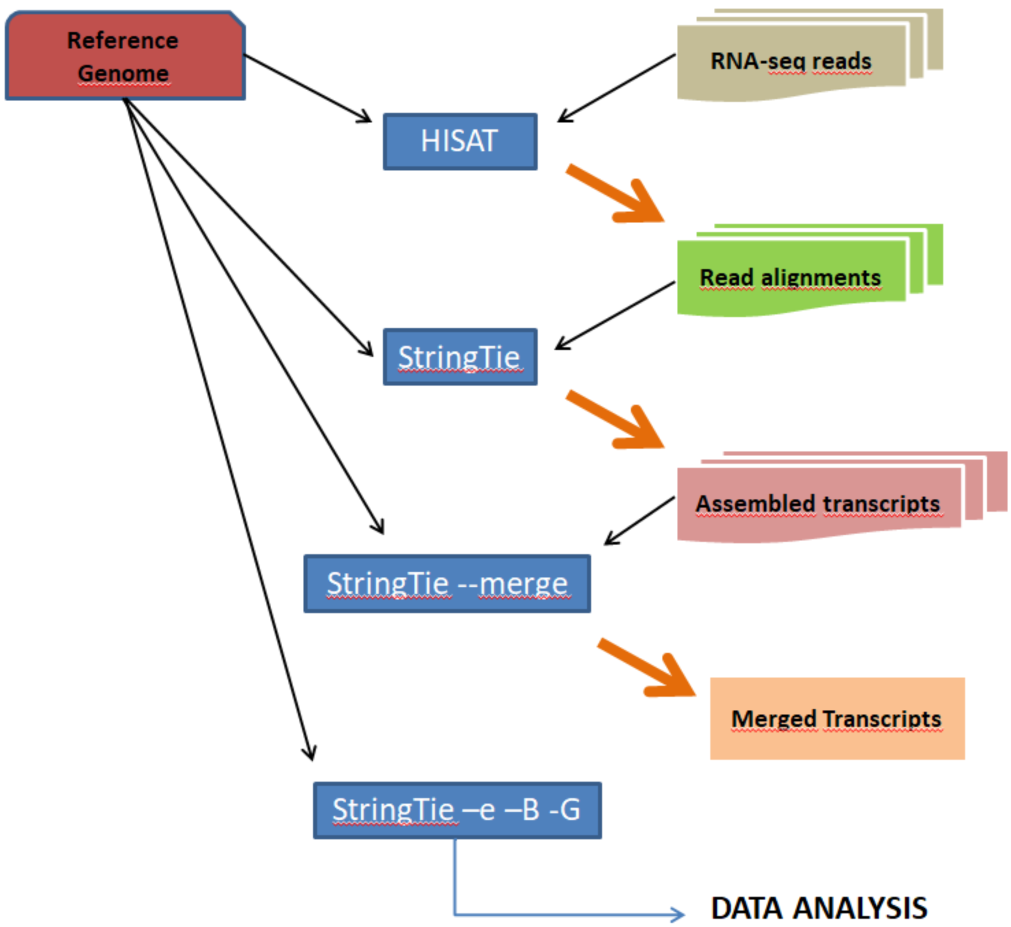


Figure 1. RNA-seq sample processing steps. Black arrows meaning data input and orange arrows meaning the output data of a function.

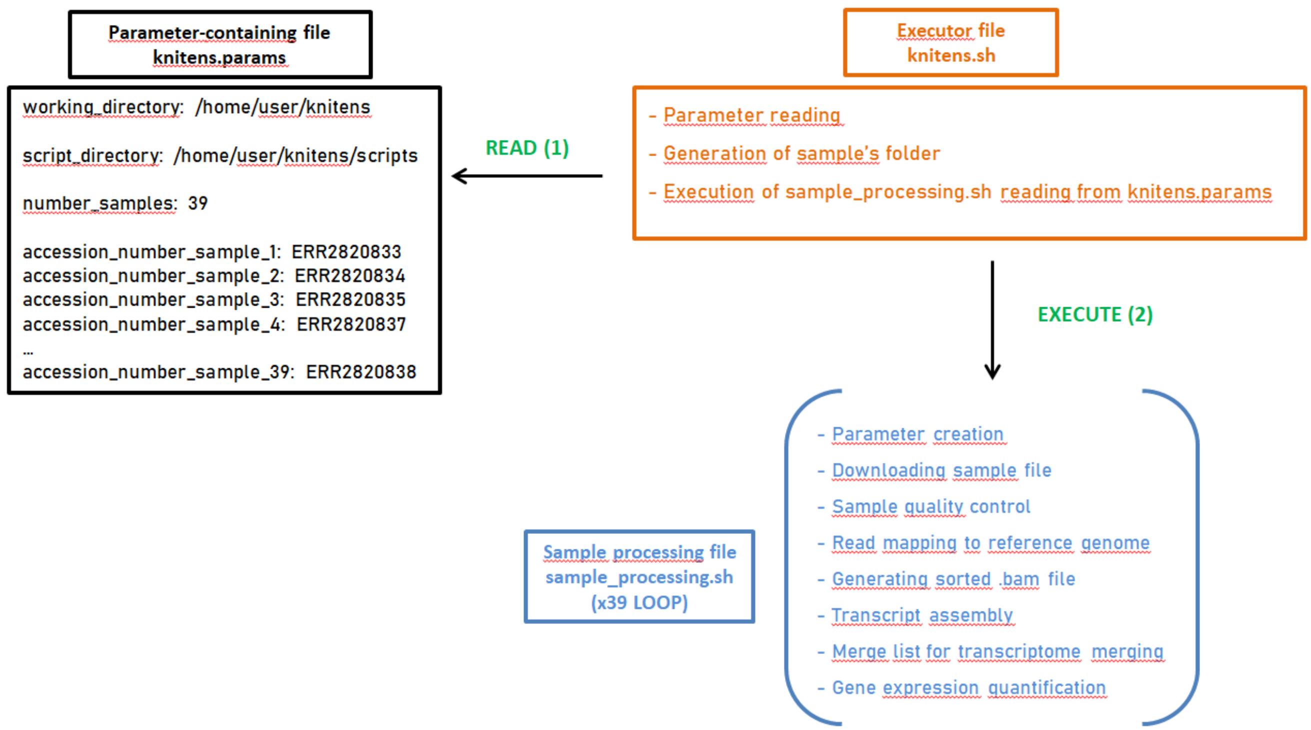


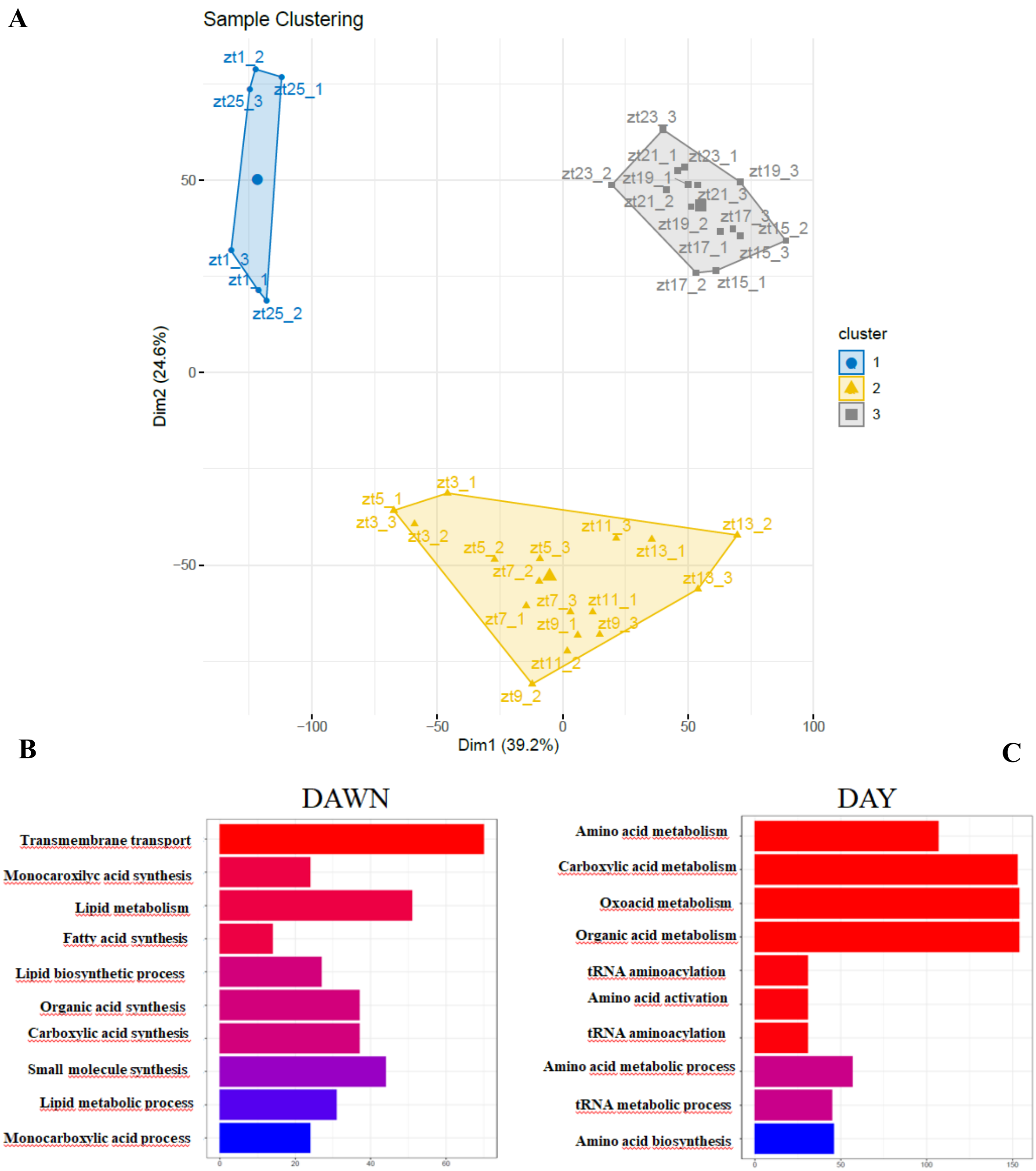
Figure 2. Bash scripting. The structure and contents of the three scripts is shown. knitens.params contains the data to be read by the executor script knitens.sh, which launchs sample_processing.sh 39 times to complete the full processing of the samples

Sample Accession Number	Number of reads	Overall alignment rate
ERR2820833	20743158	97.76%
ERR2820834	18292562	97.77%
ERR2820835	22836646	97.04%
ERR2820839	23959920	96.63%
ERR2820840	20679477	98.21%
ERR2820841	21610636	86.82%
ERR2820842	21610636	86.82%
ERR2820843	23527181	96.52%
ERR2820844	27093708	95.00%

ERR2820845	21490115	97.15%
ERR2820846	18785405	97.67%
ERR2820847	18853673	96.84%
ERR2820848	25259162	97.78%
ERR2820849	18828843	95.46%
ERR2820850	17703461	97.55%
ERR2820830	19886396	97.94%
ERR2820831	16418856	96.29%
ERR2820832	17969746	97.93%
ERR2820728	18096990	97.74%
ERR2820729	21469141	97.86%
ERR2820730	21426954	97.82%
ERR2820731	23212170	98.27%
ERR2820732	22557472	98.19%
ERR2820733	19495478	98.29%
ERR2820734	20721274	98.22%
ERR2820735	21240778	98.22%
ERR2820736	22025623	98.17%
ERR2820737	26953645	98.18%
ERR2820738	19628235	98.27%
ERR2820739	24810865	98.15%
ERR2820740	21058080	98.23%

ERR2820741	20886624	98.21%
ERR2820742	18095719	98.18%
ERR2820725	20430142	98.17%
ERR2820726	20003966	98.23%
ERR2820727	20500908	98.18%
ERR2820836	22151574	97.83%
ERR2820837	24541042	96.23%
ERR2820838	18040454	97.20%

Table 1. Number of reads for each of the 39 samples, identified by their accession numbers. The overage percentage of alignment is shown.



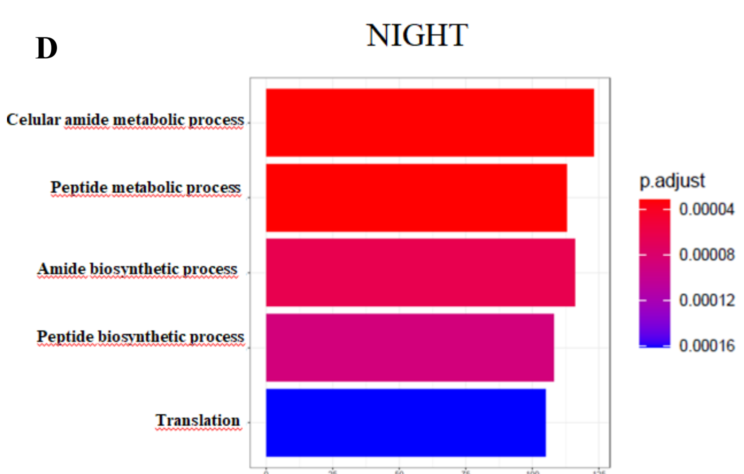
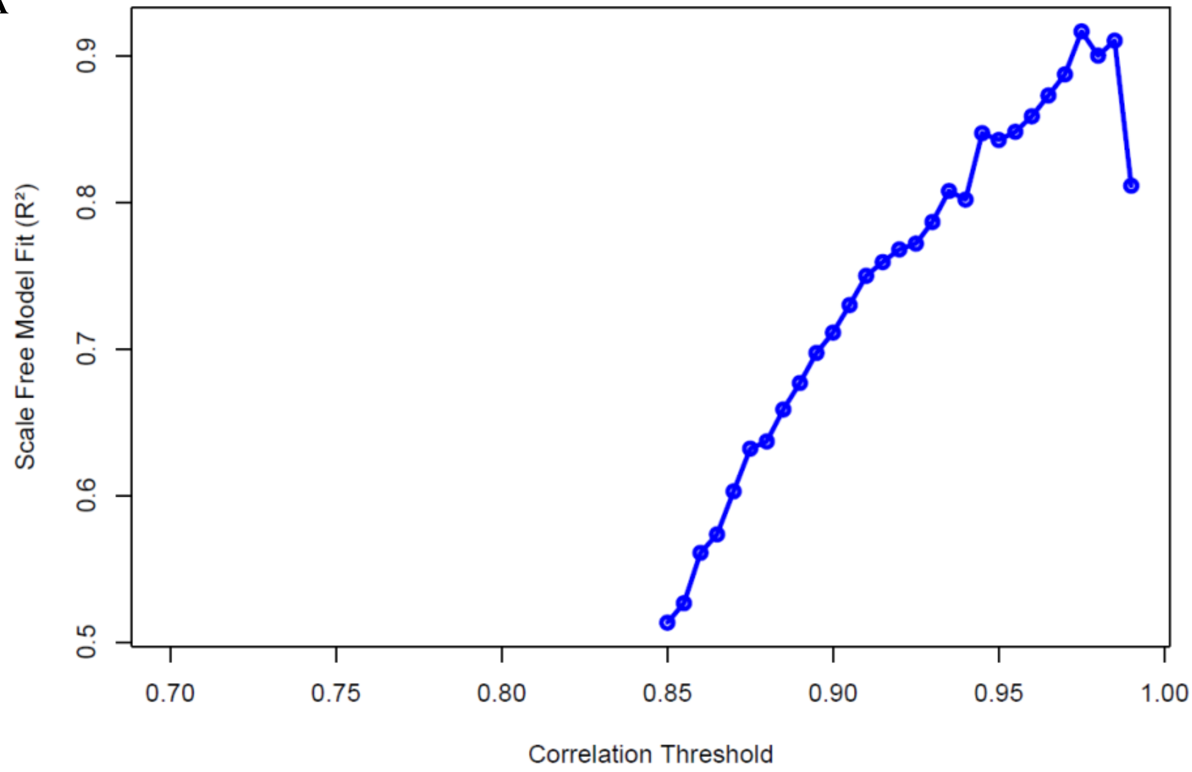
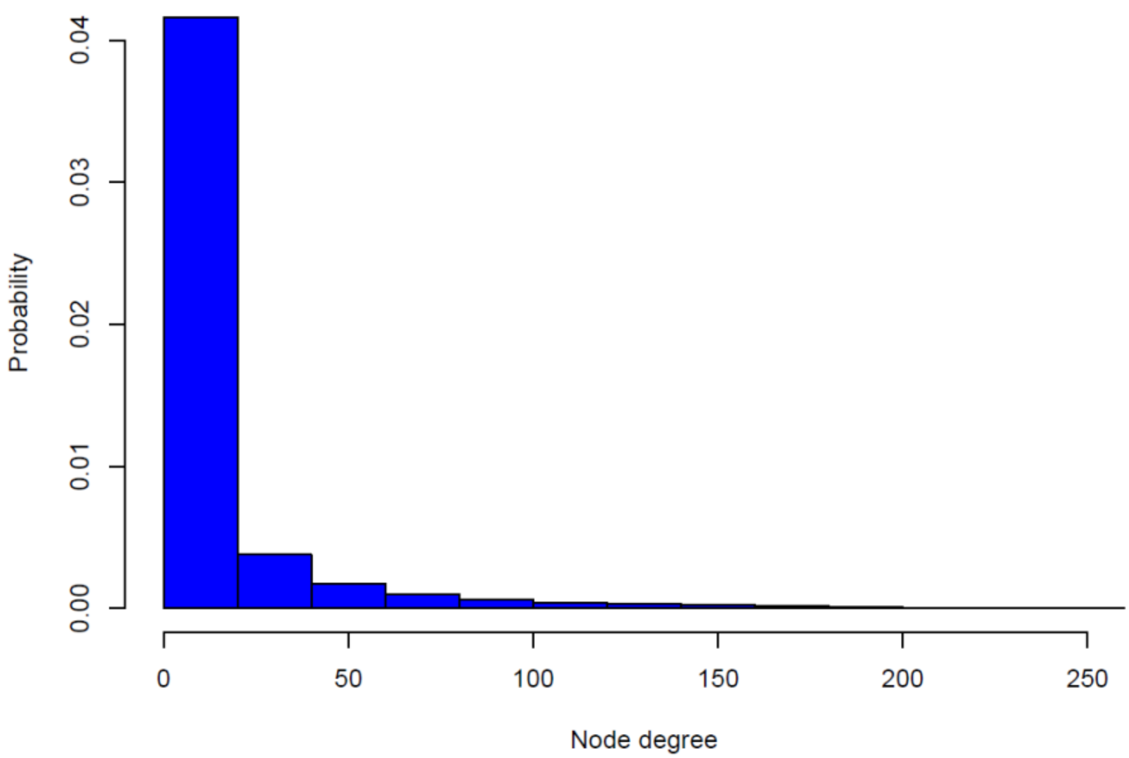


Figure 3. PCA Results. a) Clustering over PCA showing 3 different clusters forming a clock shape. We identified their correspondence to dawn, day and night over zeitgeber times. The percentage of variance explained by the two principal components is shown on the edges. b) GO term enrichment for cluster number one showing lipid and carboxylic acid biosynthesis c) GO term enrichment for cluster number two shows protein metabolism and translation d) GO term enrichment for cluster number three shows protein biosynthesis.

A


Scale Free Model Fit (R^2)
0.5
0.6
0.7
0.8
0.9
0.70
0.75
0.80
0.85
0.90
0.95
1.00
Correlation Threshold

B


Probability
0.00
0.01
0.02
0.03
0.04
0
50
100
150
200
250
Node degree

C

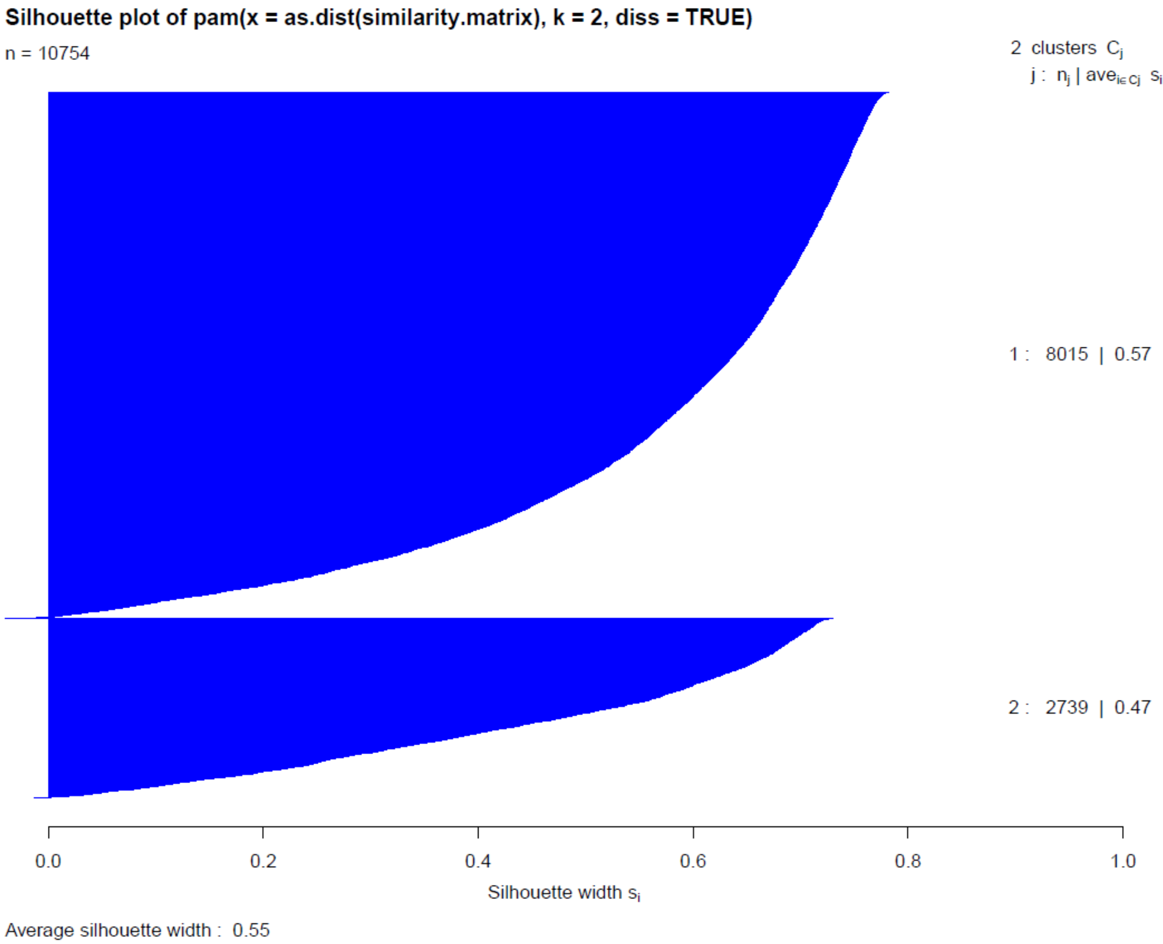


Figure 4. a) Scale free model fit vs. Correlation thresholds. The best fitting correlation value was 0.975. b) Node degree probability distribution, showing a power law distribution. c) Cluster silhouettes by PAM clustering for k = 2. Average silhouette width was maximized for 2 clusters with a value of 0.55.

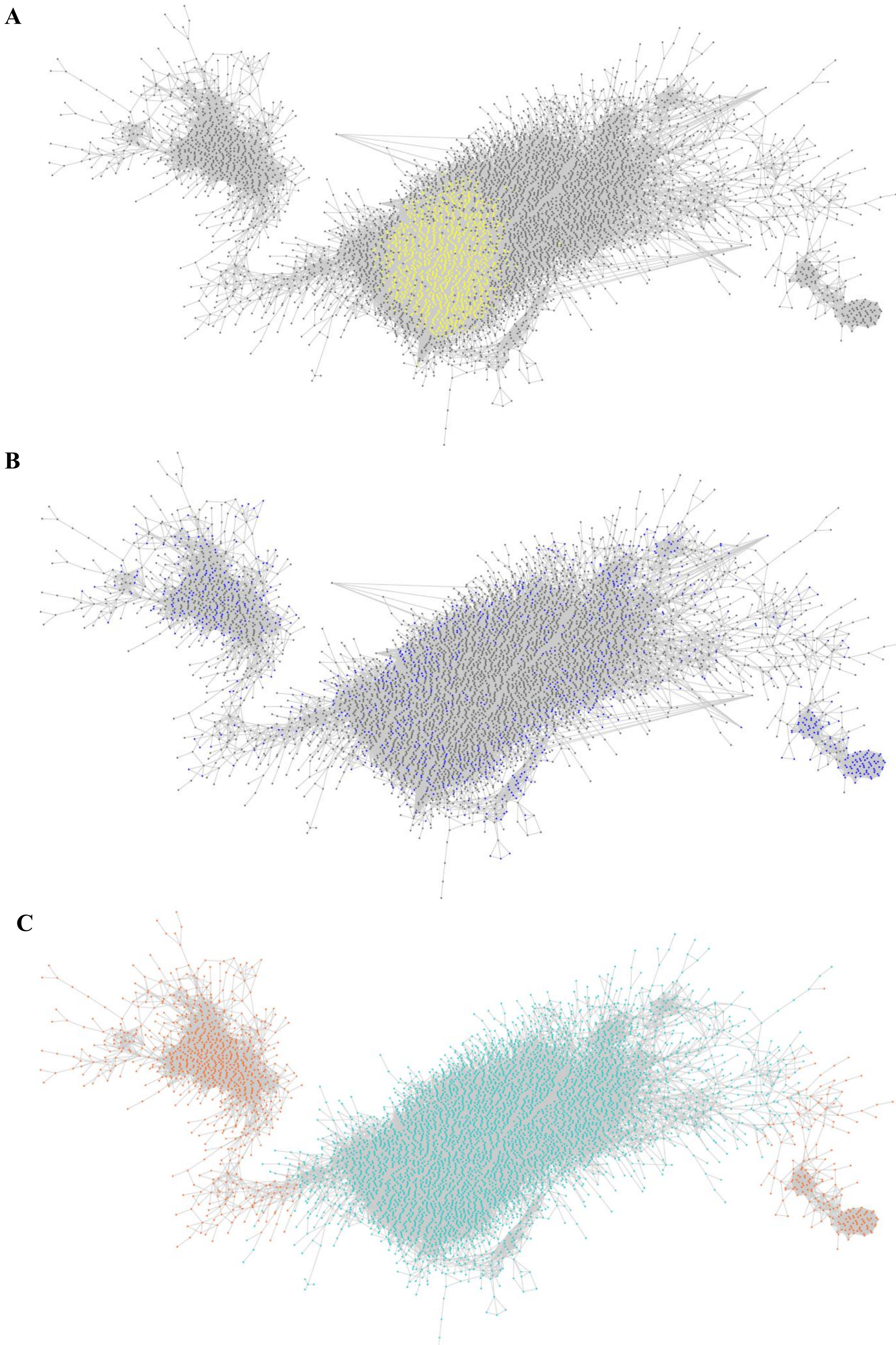


Figure 4. Network colored showing a) Network hubs in yellow. b) High Clustering coefficient nodes in blue. c) both PAM clusters in red and green.

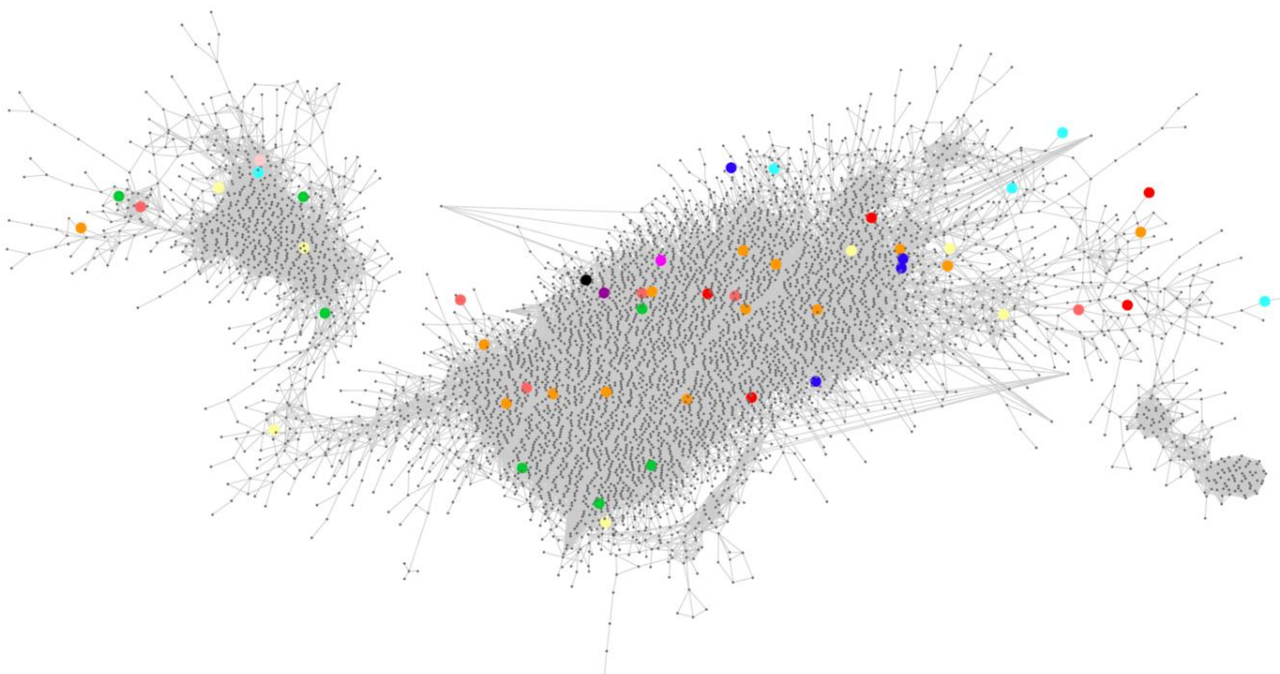
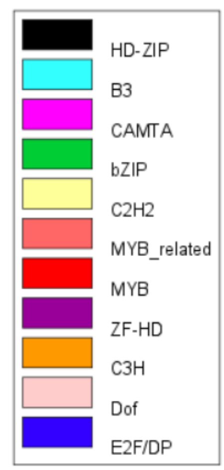


Figure 5. Location of transcription factors. We identified 90 different TFs classified into 35 families. Most of these families are not shown because they corresponded to elements that were removed because of the lack of correlation. There are four well represented families: CH3, bZIP, B3 and C2H2.



Transcriptional factor family	Functional annotation	Gene
bZIP	GO:0006396 RNA processing	kfl00031_0340 Hypothetical protein
E2F/DP	GO:0006259 DNA metabolic process GO:0006260 DNA replication GO:0051276 Chromosome organization	kfl00334_0090 E2F family transcription factor protein
GRF	GO:0006396 RNA processing	kfl00186_0090 QLQ domain containing protein)
NF-YC	GO:0006261 DNA dependent DNA replication GO:0006260 DNA replication GO:0006259 DNA metabolic process	kfl00123_0030 DNA binding histone-like transcription factor, putative
ZF-HD	GO:0006396 RNA processing GO:0034470 ncRNA processing	kfl01106_0010 Zinc finger-homeodomain protein

Table 2. Functional annotation for the most representative transcription factors found.

9 References

Burrows, M., Wheeler, D. J. (1994) A block-sorting lossless data compression algorithm, Technical Report 124, SRC (digital, Palo Alto)

Covington, M. F., Maloof, J. N., Straume, M., Kay, S. A., and Harmer, S. L. (2008). Global transcriptome analysis reveals circadian regulation of key pathways in plant growth and development. *Genome Biol.* 9:130. doi: 10.1186/gb-2008-9-8-r130

Csardi, G., Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems, 1695. http://igraph.org

Ferrari, C., Proost, S., Janowski, M. *et al.* (2019). Kingdom-wide comparison reveals the evolution of diurnal gene expression in Archaeplastida. *Nat Commun* 10, 737. https://doi.org/10.1038/s41467-019-08703-2

Fu, J., Frazee, A. C., Collado-Torres L., Jaffe A. E., Leek J. T. (2020). *Ballgown: Flexible, isoform-level differential expression analysis*. R package version 2.20.0

Hori, K., Maruyama, F., Fujisawa, T. *et al* (2014). *Klebsormidium flaccidum* genome reveals primary factors for plant terrestrial adaptation. *Nat Commun* 5, 3978. https://doi.org/10.1038/ncomms4978

Hughes, M. E., Hogenesch, J. B., Kornacker, K. (2010) JTK_CYCLE: an efficient nonparametric algorithm for detecting rhythmic components in genome-scale data sets. *J Biol Rhythm*. 2010;25(5):372–380

Jakoby, M., Weisshaar, B., Dröge-Laser, W., et al. (2002). bZIP transcription factors in Arabidopsis. *Trends Plant Sci*. 2002;7(3):106-111. doi:10.1016/s1360-1385(01)02223-3

Kiełbowicz-Matuk, A. (2012). Involvement of plant C2H2-type zinc finger transcription factors in stress responses *Plant Science : an International Journal of Experimental Plant Biology*. Apr;185-:78-85. DOI: 10.1016/j.plantsci.2011.11.015

Kim, D., Langmead, B., and Salzberg, S. (2015) HISAT: a fast spliced aligner with low memory requirements. *Nat Methods* 12, 357–360. https://doi.org/10.1038/nmeth.3317

Lance, G. N., and Williams, W. T. (1967). A general theory of classificatory sorting strategies. I. Hierarchical systems. *Comput. J.* 9, 60–64

Lê, S., Josse, J., and Husson, F. (2008). FactoMineR: An R Package for Multivariate Analysis. *Journal of Statistical Software*. 25(1). pp. 1-18

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R. (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, Volume 25, Issue 16, 15 August 2009, Pages 2078–2079, https://doi.org/10.1093/bioinformatics/btp352

Luo., Weijun., Brouwer., Cory. (2013). Pathview: an R/Bioconductor package for pathway-based data integration and visualization. *Bioinformatics*, 29(14), 1830-1831. doi: 10.1093/bioinformatics/btt285

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., Hornik, K. (2019). cluster: Cluster Analysis Basics and Extensions. R package version 2.1.0

Michael, T. P., Mockler, T. C., Breton, G., McEntee, C., Byer, A., Trout, J. D., et al. (2008). Network discovery pipeline elucidates conserved time-of-day-specific cis-regulatory modules. *PLoS Genet.* 4:e14. doi: 10.1371/journal.pgen.0040014

Monnier, A., Liverani, S., Bouvet, R., Jesson, B., Smith, J. Q., Mosser, J., et al. (2010). Orchestrated transcription of biological processes in the marine picoeukaryote *Ostreococcus* exposed to light/dark cycles. *BMC Genomics* 22:192. doi: 10.1186/1471-2164-11-192

Pertea, M., Kim, D., Pertea, G. *et al.* (2016) Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown. *Nat Protoc* 11, 1650–1667. https://doi.org/10.1038/nprot.2016.095

Pertea, M., Pertea, G. M., Antonescu, C. M., Chang T. C., Mendell J. T., and Salzberg S. L. (2015) StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nature Biotechnology*. doi:10.1038/nbt.3122

Romero-Campero, F. J., Perez-Hurtado, I., Lucas-Reina, E., Romero, J. M., Valverde, F. (2016) ChlamyNET: a Chlamydomonas gene co-expression network reveals global properties of the transcriptome and the early setup of key co-expression patterns in the green lineage. *BMC Genomics*. 2016;17:227. doi:10.1186/s12864-016-2564-y

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T. (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research* Nov; 13(11):2498-504

Thaben, P. F., Westermark, P. O. (2014). Detecting rhythms in time series with RAIN. *J. Biol. Rhythms*. 2014;29:391–400

Yoav Benjamini and Yosef Hochberg. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)* Vol. 57, No. 1, pp. 289-300

Yu, G., Wang, L., Han, Y., He Q. (2012). clusterProfiler: an R package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*, 16(5), 284-287. doi: 10.1089/omi.2011.0118

Zones, J. M., Blaby, I. K., Merchant, S. S., and Umen, J. G. (2015). High-resolution profiling of a synchronized diurnal transcriptome from *Chlamydomonas reinhardtii* reveals continuous cell and metabolic differentiation. *Plant Cell* 27, 2743–2769. doi: 10.1105/tpc.15.00498