HAL open science

Inferring cell cycle phases from a partially temporal network of protein interactions

Maxime Lucas, Arthur Morris, Alex Townsend-Teague, Laurent Tichit, Bianca Habermann, Alain Barrat

To cite this version:

Maxime Lucas, Arthur Morris, Alex Townsend-Teague, Laurent Tichit, Bianca Habermann, et al.. Inferring cell cycle phases from a partially temporal network of protein interactions. Cell Reports Methods, 2023, 3 (2), pp.100397. 10.1016/j.crmeth.2023.100397 . hal-04144618

HAL Id: hal-04144618

https://amu.hal.science/hal-04144618v1

Submitted on 28 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

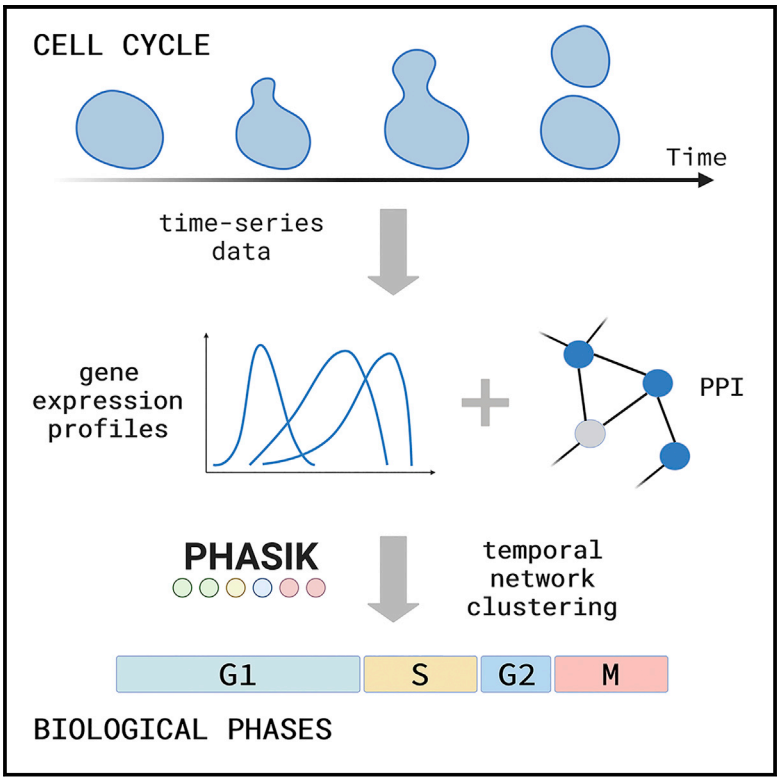
Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Cell Reports
Methods

Article

Inferring cell cycle phases from a partially temporal network of protein interactions

Graphical abstract



Highlights

- Phasik identifies temporal phases of biological systems from experimental data
- Phasik uses clustering of temporal PPI networks for biological phase recovery
- Phasik is used to infer phases of WT and mutant cell cycle and circadian rhythm
- Phasik is robust and can be used on a wide range of biological systems and data

Authors

Maxime Lucas, Arthur Morris, Alex Townsend-Teague, Laurent Tichit, Bianca Habermann, Alain Barrat

Correspondence

maxime.lucas@centai.eu (M.L.), laurent.tichit@univ-amu.fr (L.T.), bianca.habermann@univ-amu.fr (B.H.), alain.barrat@univ-amu.fr (A.B.)

In brief

Lucas et al. present a method, Phasik, that automatically identifies temporal phases and sub-phases of a biological system using interaction and high-resolution temporal expression data. They demonstrate the method and its wide applicability on different experimental datasets, including cell cycle and circadian rhythm, and assess its robustness.

Lucas et al., 2023, Cell Reports Methods 3, 100397
February 27, 2023 © 2023 The Authors.
https://doi.org/10.1016/j.crmeth.2023.100397

CellPress

Cell Reports Methods

CellPress
OPEN ACCESS

Article

Inferring cell cycle phases from a partially temporal network of protein interactions

Maxime Lucas,[1,3,4,*] Arthur Morris,[2] Alex Townsend-Teague,[2] Laurent Tichit,[1,5,*] Bianca Habermann,[3,5,6,*] and Alain Barrat[4,5,*]
[1]Aix Marseille University, CNRS, I2M UMR 7373, Turing Center for Living Systems, Marseille, France
[2]Oxford University, Oxford, UK
[3]Aix Marseille University, CNRS, IBDM UMR 7288, Turing Center for Living Systems, Marseille, France
[4]Aix Marseille University, Université de Toulon, CNRS, CPT, Turing Center for Living Systems, Marseille, France
[5]These authors contributed equally
[6]Lead contact
*Correspondence: maxime.lucas@centai.eu (M.L.), laurent.tichit@univ-amu.fr (L.T.), bianca.habermann@univ-amu.fr (B.H.), alain.barrat@univ-amu.fr (A.B.)
https://doi.org/10.1016/j.crmeth.2023.100397

MOTIVATION In many biological processes, such as the cell cycle, molecules are produced and translocated, interact with each other, and are destroyed following a strict temporal order to ensure proper execution of molecular events. An increasing amount of time-resolved data is becoming available trying to capture these ordered molecular states or phases of a biological system. To understand or predict these phases would yield crucial insight into the temporal organization of such systems.

SUMMARY

The temporal organization of biological systems is key for understanding them, but current methods for identifying this organization are often *ad hoc* and require prior knowledge. We present Phasik, a method that automatically identifies this multiscale organization by combining time series data (protein or gene expression) and interaction data (protein-protein interaction network). Phasik builds a (partially) temporal network and uses clustering to infer temporal phases. We demonstrate the method's effectiveness by recovering well-known phases and sub-phases of the cell cycle of budding yeast and phase arrests of mutants. We also show its general applicability using temporal gene expression data from circadian rhythms in wild-type and mutant mouse models. We systematically test Phasik's robustness and investigate the effect of having only partial temporal information. As time-resolved, multiomics datasets become more common, this method will allow the study of temporal regulation in lesser-known biological contexts, such as development, metabolism, and disease.

INTRODUCTION

Many biological systems go through successive phases, or states, over multiple time and space scales. Examples include biological rhythms,[1–3] cell differentiation,[4] or sleep phases in the brain,[5] just to name a few. These phases, and their order in time, are often crucial for the functioning of these systems and can even determine their fate.

A particularly important example of such a process is the cell cycle; precise timing of molecular events is crucial for proper execution of cell division. The cell progresses through 4 macroscopic phases before eventually dividing into two cells (Figure 1A). These cell cycle phases were first determined at the cellular scale by analyzing the proliferation of bean root cells.[6] The cycle starts with a first gap phase (G1), in which the cell grows and needs to reach a certain size to enter the next phase, where DNA is synthesized (S), followed by a second gap phase (G2) and finally mitosis (M), in which the duplicated chromosome set is divided and equally distributed into two daughter cells. These 4 phases can be further divided into sub-phases, or physiological processes. For example, mitosis is composed of prophase, metaphase, anaphase, and telophase.

In the past 70 years, extensive cell cycle research identified the molecular events driving the behavior of the cell in each of these 4 phases (see reviews by Koch and Nasmyth,[7] Murray,[8] Nasmyth,[9,10] and Vodermaier[7–11]). In particular, the transcriptionally controlled cyclins and the cyclin-dependent kinases were found to regulate a multitude of cell cycle events. A cellular destruction machinery complements these proteins by degrading cell cycle regulators at specific times in the cell cycle. A set

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

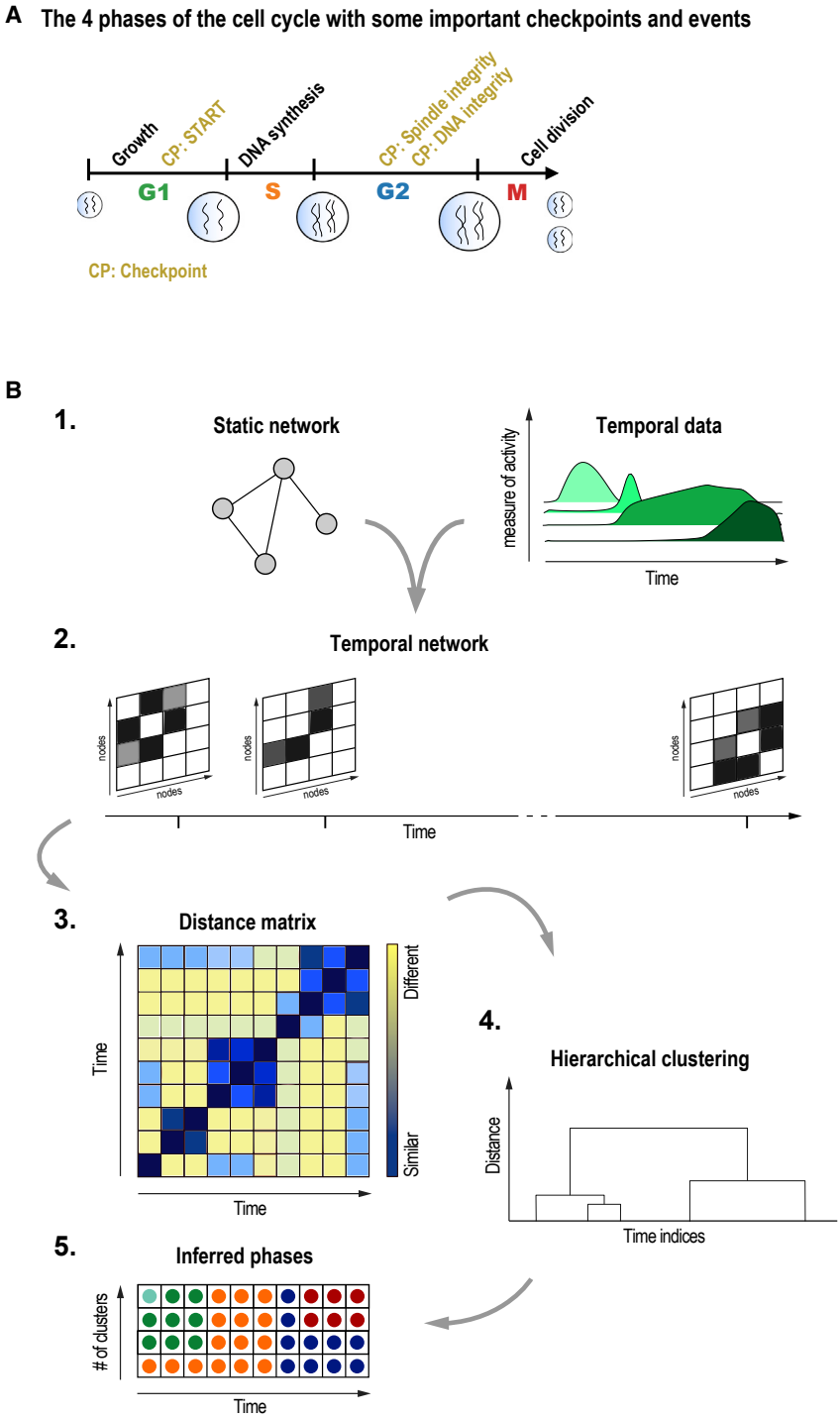


Figure 1. Schematics of the temporal system of the cell cycle and how to extract its phases using temporal network clustering

(A) The cell cycle consists of a succession of physiological processes mediated by PPIs, leading to division of the cell. The ordering of these interactions is crucial and is ensured by molecular checkpoints (yellow). The cycle is divided into 4 main phases: Gap1 phase (G1), synthesis phase (S), Gap2 phase (G2), and mitosis (M).

(B) Multiscale phase inference with Phasik: schematic of the method. (1) Edge time series are integrated into a static PPI network to build (2) a partially temporal network, shown in snapshot representation. (3) Pairwise distances between snapshots are used to (4) cluster snapshots hierarchically. (5) The clusters obtained: each row corresponds to a fixed number of clusters. Each snapshot is shown as a dot whose color represents the cluster to which it belongs. Clusters can be interpreted as time intervals because there is a 1:1 correspondence between snapshots and time points. Each cluster can then be interpreted according to the underlying biological processes.

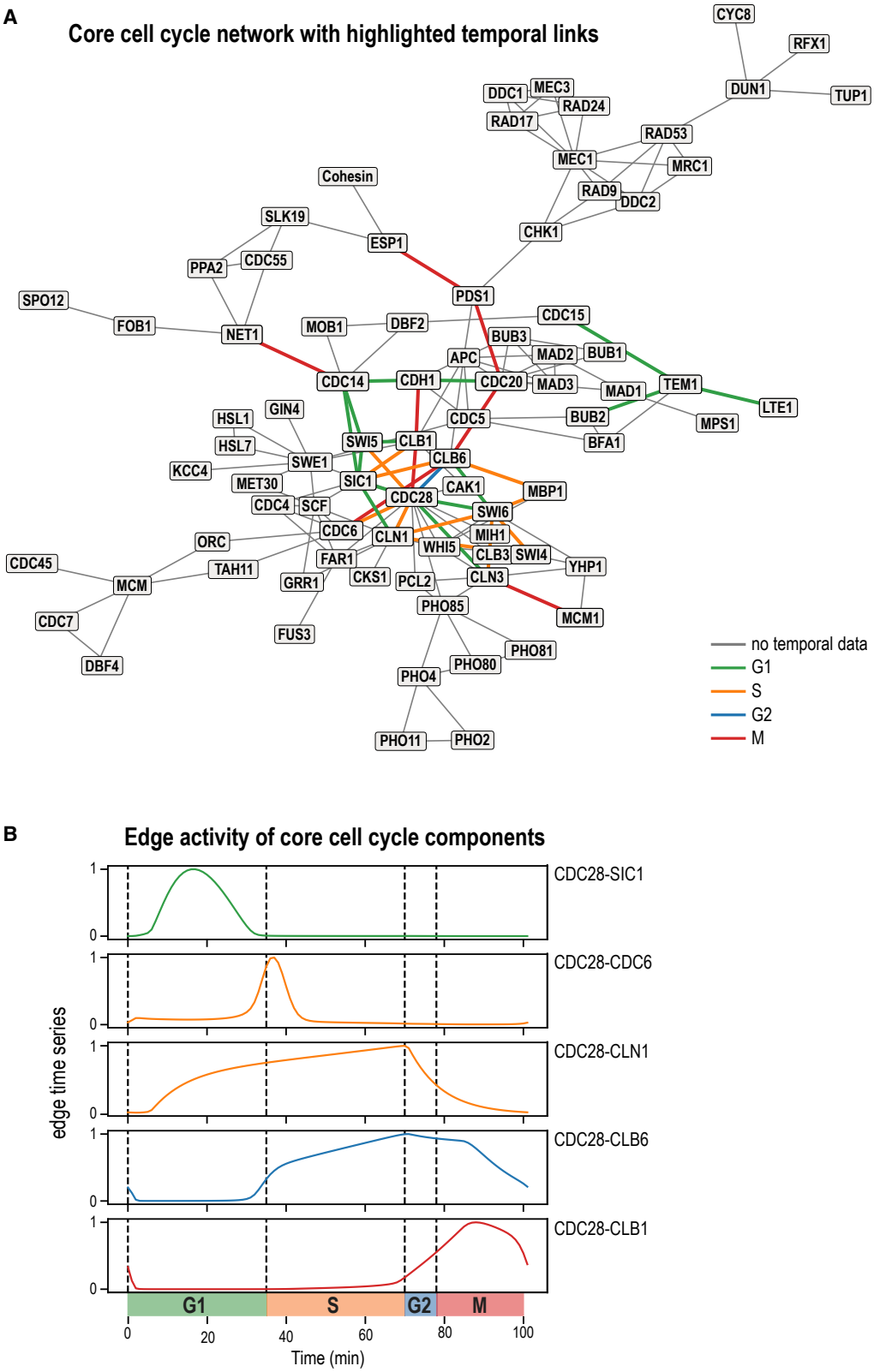
of core cell cycle proteins thus regulates the cell cycle and its phases by interacting with each other in a precisely timed manner. These interactions can be of different types, stable and long term or transient (leading, for instance, to protein phosphorylation or ubiquitination), or represent a regulatory relationship. They drive progression of the cell cycle and ensure that one phase is complete before the cell enters the next phase by acting as so-called molecular checkpoints. Not fulfilling a checkpoint can stop progression of the cell cycle and change the fate of the cell.

The complexity of interactions between proteins is often encoded in a network representation.[12] In such biological networks, nodes represent proteins, and edges represent protein-protein interactions (PPIs).[13,14] The cell cycle, for instance, has been represented as a network of PPIs in the Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway resource.[15] These network representations are, however, static; they do not include any information about the time-varying interactions between cell cycle regulators.

Nonetheless, over the past few years, temporal information about biological systems has become more widely available in the form of time series recordings thanks to high-throughput techniques such as RNA sequencing and high-throughput proteomics. This has opened the door to investigation of the temporal dynamics of various systems. In addition, mathematical models of specific biological systems have been proposed, based on *a priori* knowledge. These models typically describe the dynamics of gene expression levels[16] or protein concentrations,[17] either as continuously evolving variables or using binary values.[18]

To incorporate such temporal information into the representation of PPIs, a natural framework is that of temporal networks.[19,20] In a temporal network of PPIs, nodes represent proteins, and

CellPress
OPEN ACCESS



(legend on next page)

interactions between them are represented by time-varying edges. Temporal network theory has been used successfully in areas ranging from social interactions[21–23] to neuroscience.[24,25] However, despite calls to use it more in biology,[26] it is still underused to study, e.g., PPI networks. Existing studies mostly integrate gene expression data from microarrays or RNA sequencing to a static PPI network[27] to identify active subnetworks[28] and functional modules.[29–32]

Here, we go further and take full advantage of the temporal network framework. We developed an automated pipeline to investigate the temporal organization of the cell cycle across a range of timescales, starting from time series data of PPIs. To this end, we built a representation of the cell cycle as a temporal network of PPIs by incorporating time series data, obtained from a mathematical model of the cell cycle, into a static network of PPIs. We then built on a recent method introduced by Masuda and Holme[33] to infer phases by clustering snapshots of the temporal network. We finally validated these phases against ground truth biological knowledge. For this, we chose to apply our pipeline to infer the phases of the cell cycle of budding yeast, one of the best-described cyclic systems to date. Within this pipeline, we present in particular three main methodological advances: (1) we analyze the phase inference results at multiple temporal scales, revealing several additional sub-phases in the cell cycle, (2) we systematically analyze the robustness of the results with respect to changes in the clustering methods and noise, and (3) we investigate the effects of missing or partial temporal information on the detection of phases because a number of models or biological datasets contain temporal information only on subsets of the proteins or genes of interest. In addition, we show that biological phases can also be inferred by using gene expression time series instead of temporal PPI data to build the temporal network. Finally, to test our method and demonstrate its general applicability, we applied it to identify phase arrests of cell cycle mutants in yeast. We also analyzed temporal data of the circadian rhythm in the mouse liver, in wild-type as well as *Per1/2* and *Bmal* knockout mouse models, and identified the circadian rhythm in the wild type and an ultradian rhythm in the knockout mouse liver.[34] We made our code user friendly and readily available for others to use for other biological systems (https://gitlab.com/habermann_lab/phasik).

RESULTS

The Phasik workflow

The workflow of our analysis consists of two main steps: (1) building a temporal network representation of the cell cycle and (2) inferring cell cycle phases from it. This workflow, which we call Phasik, is illustrated in Figure 1B. We first describe this general workflow in the specific case of the yeast cell cycle using time series from an ordinary differential equation (ODE) model. This well-known system allows us to show that we can correctly recover known biological results.

Building a temporal network representation of the cell cycle

To build a temporal PPI network that represents the yeast cell cycle, we combined a static PPI network built from KEGG pathway data[15] (Figure 2A), with temporal interaction data based on the mathematical model of the cell cycle as described by Chen et al.[17] (Figure 2B). The static PPI network consists of 83 proteins (nodes) and 159 PPIs (edges). To obtain a temporal network, we first acquired time series of protein concentrations by numerical integration of the ODE model of Chen et al.,[17] with a time step of 1 min over a full cycle of 101 min. For each edge A-B between proteins A and B in the static PPI network, we then defined its weight $w_{AB}(t)$ at time t as the product of the concentrations of the two proteins to quantify their co-presence at that time.[35] Such a time-varying weight could be defined only for the edges connecting two proteins, which are described by the ODE model. In addition, the ODE model contains a few special variables representing the concentration of protein complexes; these quantities could also be directly used as evolving edge weights for the edges between the proteins forming these complexes (STAR Methods). Finally, we normalized each weight time series by its maximum value over the cell cycle so that all weights vary between 0 and 1.

Because the ODE model provides overall temporal information only for a subset of nodes and, thus, edges in the static PPI network, the resulting representation is a partially temporal network containing static edges for which no temporal information is available and temporal edges whose weights evolve in time. The weights of the static edges are set to a constant value of 1.

This procedure yielded time-varying weights for only 34 of the 159 edges (21%) of the PPI network. Several examples of the time series of edge weights obtained from the ODE model are shown in Figure 2B (Figure S1A shows all time series, and Video S1 shows an animation of the temporal cell cycle network), and the temporal edges are highlighted with colors in Figure 2A (with the color representing the phase during which the weight is largest).

Overall, the PPI temporal network can be seen as a series of n network snapshots (Figure 1B), each corresponding to a time step at which the edge weights are observed. The time between successive snapshots equals the temporal resolution used in the ODE numerical integration. Importantly, this general procedure can also be used to build a temporal network from other types of data, such as gene expression data from RNA sequencing (RNA-seq), and for other biological systems, as will be demonstrated below.

Inferring phases from a temporal network

Because biological phases and processes are driven by specific PPIs, it is expected that each phase could be related to a specific structure or "state"[25,33] of the temporal PPI network. A large similarity between snapshots of the temporal network

Figure 2. Partially temporal PPI network of the cell cycle

(A) Static representation of the temporal network, containing 83 nodes and 159 edges. For visualization purposes, the 34 temporal edges are shown in a color that represents the phase of their peak activity. Edges lacking temporal information are shown in gray.

(B) Edge times series of core cell cycle PPIs.

CellPress
OPEN ACCESS

at different times could indicate that the system is in the same phase at these times, and low similarity between successive times could indicate a change of phase.[21,25,33]

This idea can be taken further and formalized to infer phases by performing a clustering of the snapshots of the temporal PPI network. Such inference involves three steps,[33] illustrated in Figure 1B. Given a temporal network, (1) we compute the distances between each pair of snapshots and encode them in a distance matrix; (2) using this distance matrix, we apply a (hierarchical) clustering algorithm on the snapshots, and (3) we extract the clusters after choosing the desired number of clusters. Each cluster of snapshots can then be traced back to the set of time steps corresponding to those snapshots and interpreted as a phase of the cycle (STAR Methods). Finally, when the corresponding biological knowledge is available, the inferred phases can be compared with ground truth knowledge of the cell cycle and interpreted biologically. In less known systems, ground truth is not always available, and our method might yield new insights and directions for experimentalists to test. In both cases, it is possible to see which PPIs are most (in)active during each phase and potentially bring new insight too.

In most clustering algorithms, the number of clusters is fixed *a priori*. However, many real systems, including biological ones, exhibit dynamics on multiple timescales. To explore a range of timescales and, thus, the potential hierarchy of phases and sub-phases of the cell cycle, we compute clusterings with varying numbers of clusters (Figure 1B); the main cell cycle phases are discovered at a small number of clusters, whereas sub-phases require more clusters. For example, computing 2 clusters gives us a more coarse-grained version of the temporal organization than computing 5 clusters.

The phases determined by the clustering algorithm, at a fixed number of clusters, may be affected by two main technical choices in the pipeline: (1) the distance measure used to compute the distance matrix between snapshots and (2) the specific clustering algorithm considered together with its parameters, in particular how the distance between clusters is measured. Unless stated otherwise, we used (1) the Euclidean distance and (2) hierarchical clustering with the "Ward" method of computing the distance between clusters in our analysis. These settings were chosen as a reference because they provided the best clusters according to the following criteria: biological interpretability, quality of clustering, and robustness of the method.

We used the average silhouette score to assess the quality of each clustering.[36] This score is a number between -1 (the worst) and 1 (the best) that indicates how well separated the clusters are; values near 1 indicate very good separation (hence a meaningful partition of the snapshots), while values near 0 indicate overlapping clusters. Negative values indicate that one of the snapshots was not correctly assigned to a cluster. In addition, we quantitatively investigated the robustness of the clusterings (at a fixed number of clusters) with respect to the parameter choices by computing similarity scores between clusterings. To do this, we used the adjusted Rand index, which can take values between -1 and 1. A value of 1 indicates that two clusterings are identical (up to permutation of the labels), and a value of 0 indicates random cluster assignments. Negative values can occur when two clusterings are less similar than expected for random cluster assignments (STAR Methods).

Inferring the multiscale phases of the cell cycle

We inferred phases of the budding yeast cell cycle from the temporal cell cycle network, illustrated in Figure 2 (animated in Video S1), using the Phasik workflow described above. First, we computed the distance matrix, as described above for snapshots observed at a temporal resolution of 1 min (Figure 3A). A visual inspection of this matrix revealed prominent dark blue diagonal blocks corresponding to highly similar successive snapshots of the temporal network, thus indicating a marked temporal structure of the system. Second, we applied hierarchical clustering and computed clusterings ranging from 2–11 clusters. The resulting partitions of the cell cycle timeline are shown in Figure 3B. Figure 3C displays the average silhouette that measures the quality of each clustering as a function of the number of clusters. This quality score was around 0.5–0.6 in all cases, indicating that no partition should be discarded from this criterion. In fact, the roughly constant value of the silhouette rather indicates that many timescales are relevant in the system. The largest values of the average silhouette were obtained for numbers of clusters ranging from 5–9.

To validate the results of the Phasik pipeline, we took advantage of the biological knowledge available for the budding yeast cell cycle and checked that the starting and ending times of each cluster corresponded to the timings of known biological phases and events. We used the timings of the primary cell cycle phases G1, S, G2, and M, determined relative to the budding yeast cell modeled in Chen et al.[17] In addition, we used the timings of the following checkpoints and physiological events as they were described in the same study: "bud," indicating bud emergence; "ori," indicating the start of DNA synthesis; "spn," indicating completion of chromosome attachment to the spindle and alignment of chromosomes on the metaphase plate; and "mass," indicating the start/end of a cycle. Timings of the START checkpoint and the "E3" event (a short excitation period associated with a sharp decrease in SBF [SCB-binding factor] complex concentration) were determined by a stability analysis in Lovrics et al.[37]

Visual inspection of the distance matrix of the temporal cell cycle network, returned by Phasik, indicated 4 major clusters roughly corresponding to the G1, S, G2, and M phases of the cell cycle (Figure 3A) as well as a small cluster at the very beginning of the cycle. Application of the clustering algorithm allowed us to perform a more detailed and precise analysis of the clusters at each number of clusters (Figure 3B) and to discuss them in relation with the evolution of the edge weights shown in Figures S1A. For 2 clusters, the algorithm detected a phase corresponding to the G1/S phases, starting at the START checkpoint, when the cell commits to entering the cell cycle, and ending at the E3 excitation peak before entering the next phase, and a second phase corresponding to the G2 and M phases of mitosis, which persists until the START checkpoint. The main determinants for this clustering are the abrupt increase (START) and decrease (E3) of the activity of edges SWI4-SW6 and SWI6-MBP1. These two edges correspond molecularly to the activity of the SBF and MBF (MCB binding factor) complexes, respectively, which are known to initiate at the START

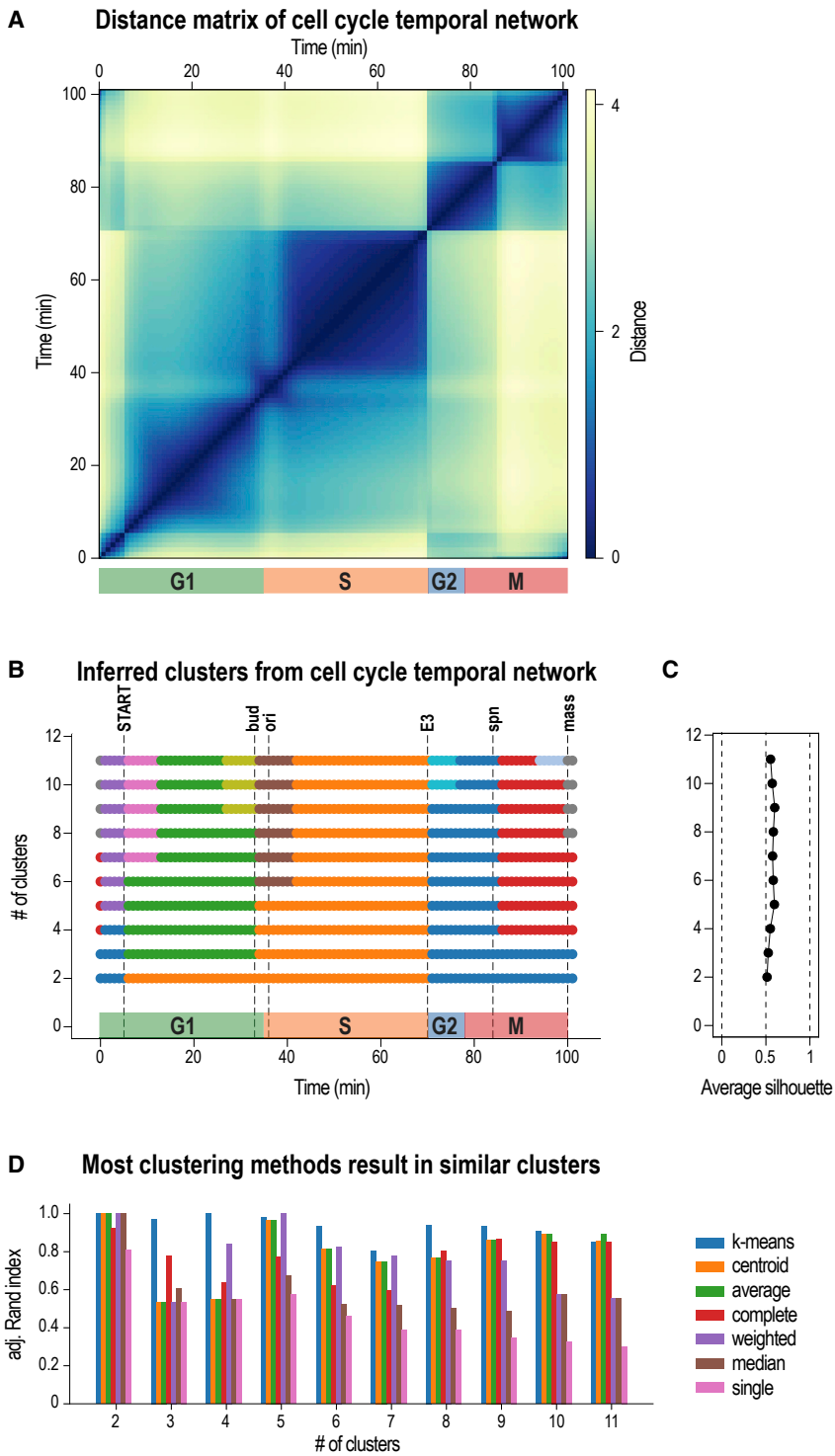


Figure 3. Cell cycle phases inferred for a range of timescales from the temporal cell cycle PPI network

(A) Distance matrix using Euclidean distance.

(B) Clusters inferred for 2–11 clusters.

(C) Quality of clustering for clusters in (B), quantified by their average silhouette. Values close to 1 indicate well-separated clusters. Similar values of about 0.6 indicate that no number of clusters can be readily discarded.

(D) Most clustering methods resulted in clusters similar to the "Ward" method used in (B) and (C), as shown by the adjusted Rand indices.

checkpoint. They are known to be activated by the CDC28-CLN3 complex early in the cycle and are required for activation of gene expression of cyclins CLN1 and CLN2.

For 3 clusters, the separation between phases G1 and S was detected at the physiological event of "bud". Several molecular events can be linked to this separation: emergence of the CDC28-CLB6 and SWI6-CLB6 edge weights, which persist until the E3 event, as well as short peaks of the weights of CDC28-CDC6 and CDC6-CBL6. Moreover, the weight of CLB6-SIC1 decreases. The weights of CLN1-SIC1 and CDC28-SIC1 seem to be restricted to the G1 phase.

For 4 clusters, the four phases G1, S, G2, and M were clearly detected, although the transition of the G2/M phases was obtained at the "spn" checkpoint. This checkpoint coincides with the abrupt change in edge weights for BUB2-TEM1, CDC15-TEM1, CDC20-CLB5, CDC20-PDS1, as well as LTE-TEM1. Biologically, it represents the moment in metaphase when all chromosomes should be properly attached in a bipolar fashion to the mitotic spindle. When the spindle checkpoint is passed, the cell progresses into anaphase. Furthermore, the beginning of the G1 phase (before START) was clustered with the G2 phase, indicating that they are more similar to each other than to other parts of the cell cycle. However, this split was not robustly found with other clustering methods (also discussed below).

For 5 clusters, the 4 phases G1, S, G2, and M were identified, with the G1 phase split in a pre-START (purple in Figure 3B) and post-START (green) cluster. The cluster corresponding to G2 extends to metaphase and the "spn" checkpoint.

For 6 clusters, a small (brown) cluster appeared at the transition between phases G1 and S, immediately after the "bud" event, corresponding to a peak in CDC28-CDC6 and CLB6-CDC6 edge activities. CLB6 is known to be required for initiation of DNA synthesis by activating CDC28, as well as the DNA replication protein CDC6. Both interactions are thus initiating the transition from phase G1 to S.

CellPress
OPEN ACCESS

For 7 clusters, a new cluster (pink) was detected in G1, right after the START checkpoint. Several changes in edge activities can be linked to this new cluster; CDC14-SIC1 and SIC1-SWI4 have sharp activity peaks at this cluster. Furthermore, the edge activities of BUB2-TEM1, CDC14-CDH1, CDC14-SWI5, CDC15-TEM1, CDC20-CDH1, and CLB1-SIC1 drop during this phase, and CDC14-NET1 shows a subtle increase. Finally, there is the transition of MBP1-SWI6, SWI4-SWI6, CDC28-SIC1, CLN1-SWI4-CLN1-SWI6, and CLN1-SIC1 from low to high activity contributing to this new cluster.

For 8 clusters, a new cluster (gray) was introduced at the end of the M phase, which extends to the beginning of G1. It can be linked to the late edge activities of CHD1-CLB1 and CDH1-SIC1, well agreeing with the APC-CDH1 complex degrading CLB1 to induce mitotic exit. Furthermore, the edge activity of CLB1-SWI5 and CDC28-SWI5 drops.

For 9 clusters, a new cluster (light green) was found toward the end of G1 right before the "bud" event, coinciding with the short and prominent peak in the edge activity of CLB6-SIC1 (Figure S1A). Biologically, CLB5/6 are already prominently expressed at this stage and have formed stable complexes with CDC28. However, until the onset of S phase, they are inhibited by SIC1. Following destruction of SIC1, replication is initiated.

For 10 clusters, G2 phase (light blue cluster) was properly separated from M phase at the E3 excitation point. More subtle edge activity changes most likely contribute to this new cluster: the transition from high to low CDC28-CLN1, the subtle rise in CLB1-SIC1 and CLB6-SIC1, and the rise in CLB1-SWI5, CDC28-SWI5.

For 11 clusters, mitosis was further refined, with detection of a separate anaphase (red cluster) and telophase (light gray cluster). This new split coincides with changes in the activity of edges CDC14-NET1, CDC14-SWI5, CDC20-CLB6, CDC20-PDS1, CDC20-CLB1, and ESP1-PDS1.

Overall, the computed clusters correspond to known biological processes across a range of timescales, and these clusters were of a high quality according to their silhouette scores. Note that, for high numbers of clusters, the clustering obtained presents much more finely time-resolved pictures of the temporal organization of the cell cycle than with the major cell cycle phases, with biological events happening over shorter timescales. A more detailed view of the silhouette scores is shown in Figure S1B for each case. For 6 or more clusters, negative silhouette values appeared for a few snapshots. Because the average silhouette score was highest at 5 clusters, and because these seem to be the most biologically relevant for detecting the basic phases of the cell cycle, we use this cluster number in the rest of this study.

Testing the robustness of Phasik

We tested the robustness of Phasik's results against two big classes of perturbations: changes in Phasik's method itself (temporal network construction, clustering methods) and changes in the input data (measurement noise and sampling frequency). We outline here the main results of these tests and provide all details in Methods S1. Overall, we found that Phasik's results were robust.

First, we tested how the normalization of the edge weights from the ODE model affected the results. We found that not normalizing edge weights led to less relevant clusters (see Figure S2 for details). This is likely to be due to large differences in protein concentrations (up to a factor of 200). Concentration changes of proteins with low concentrations, which might be biologically important, thus have virtually no effect on the clusters. It therefore seems to be important to normalize edge weights.

We next tested the robustness against variations in the clustering method. Our base method was hierarchical clustering with Ward linkage. We found similar clusterings for hierarchical clustering with six other linkage methods as well as with K-means (Figures 3D and S3A–S3G). We also compared three distance metrics with our base Euclidean distance; all three gave nearly identical clustering at each cluster number (Figures S3H–S3L). Overall, identifying similar clustering with different clustering methods is an indication that the data contain strong phasic information.

Next, we tested the influence of experimental noise on the results because experimental data come with noise. To simulate measurement noise, we added Gaussian noise to the original edge time series before applying the Phasik pipeline. We did so for a range of noise strengths and three numbers of clusters (2, 5, and 10). The results were robust against weak noise but dropped in quality above a threshold noise strength (Figure S4). Moreover, this threshold was higher for lower numbers of clusters, making them more resistant to noise than clusterings with more clusters.

Finally, the temporal resolution of experimental data is often limited, contrary to that obtained from simulating ODEs. To test the effect of coarser time resolutions, we artificially down-sampled the edge weight time series from our base resolution of 1 min to a range of coarser resolutions with a maximum of 20 min. We found that clusters were also in good agreement with known phases for coarser time resolutions (Figures 4A, 4B, and S5). Coarser time resolutions, however, yielded lower silhouette scores on average and fewer timepoints to cluster. Combined, these effects suggest a lower bound on acceptable time resolutions that will depend on the timescales of interest in the system considered.

How much temporal information is needed to infer meaningful phases?

Temporal networks are usually considered when temporal information is available for all pairs of nodes, and all edges are then temporal. Here, instead, temporal information is available only for 21% of the edges, and we set a constant weight to the other edges of the PPI. Such a situation of partial temporal information might in fact happen in many systems, in particular in biology, where experimental measurements of temporal information about all interactions are often not feasible.

We thus investigated how detection of meaningful phases with Phasik depends on the amount of temporal information available. We artificially discarded the temporal information from selected edges, building temporal networks with increasingly lower temporal information. We then computed clusters from these networks and compared them with those obtained with all available temporal information.

Discarding temporal edge activity for each specific edge resulted in most cases in nearly identical clusters (Figures S6A–S6C). We thus focused on nodes rather than single edges to

investigate whether some nodes are more important for extracting meaningful phases (i.e., they are more temporally central). This test is especially meaningful because edge weights were defined based on time series related to nodes. We built temporal networks using temporal information restricted to one specific node and its interaction partners, setting all other edge weights to a constant value (illustrated in Figure 4C). We then computed the clusters using Phasik with 5 clusters and compared the result with the one of the whole temporal network using the adjusted Rand index (Figure 4D). We performed this procedure for each node present in the ODE model. In some cases (e.g., CLB6, CDC28, or CLN3), the similarity to the original clustering remained high. For some other nodes, however (e.g., CDH1, CDC14, CDC6), very different clusters were obtained (Figure 4D).

More details on the distance matrices and clusters obtained are shown in Figures S6D–S6H. When temporal information was restricted to the edges connected to CDC28, the original 5 clusters were reproduced fairly well (Figure S6D). Likewise, keeping only temporal edge weights for MBP1 yielded clusters quite similar to the original ones (Figure S6E). Examples of nodes whose sole temporal information recovered the original 5 clusters less well included CLB1, which detected the G2 and M phases better but was unable to detect the G1 and S phases (Figure S6F), as well as SIC1, which detected only one cluster over the S, G2, and M phases, whereas G1 was split into 3 distinct clusters (Figure S6G).

We finally investigated whether some features of the nodes could predict their importance in this procedure, as determined by their possibility to recover clusters similar to the original ones when only their activity timeline and the ones of their partners are known (Figure S6H). We found only very weak correlations between the adjusted Rand indices in Figure 4D and the number of edges with temporal information with which a node participates. Similarly, only very weak correlations were found with typical measures of centrality for static networks, such as degree or betweenness centrality, with several nodes having low centrality but a high adjusted Rand index. Only the eigenvector centrality showed a slightly larger (but still weak) correlation with the adjusted Rand index.

Inferring modified cell cycle phases in mutants

In cell cycle mutants, phase behavior is changed, and some mutant cells stop their cell cycle progression in one of the phases, for example, G1 (referred to as G1 arrest). Here, we wanted to test whether Phasik can detect these modified phases. We simulated mutant time series from the same ODE model by changing the value of a few key parameters for two mutants: a CLN1/2/3 knockout and a CLB1/2 knockout, which display G1 and G2 arrest (in the ODE model, and experimentally), respectively. We then built a temporal network with the time series of each of these mutants and computed the associated clusters.

In the CLN1/2/3 mutant (Figures 5A and 5B), Phasik detected one main cluster (orange) that starts during G1 and lasts until the end of the cycle at lower cluster numbers where the average silhouette is high. This corresponds to the expected G1 arrest. Note that a later cluster (purple) appears at 5 and more clusters. This cluster appears because of normalization of the edge time series, which amplifies the very small variations of the quasi-constant unprocessed time series (see, e.g., SW4-SWI6 in Figure S7). Clustering the temporal network built from the non-normalized time series does not yield this additional late cluster (data not shown). In the CLB1/2 mutant, we detected a slightly prolonged S phase and, at lower numbers of clusters, one main cluster (blue) stretching from S phase to the end of the cycle (Figures 5C and 5D). This large cluster persisted up to 6 clusters, at which point the silhouette score dipped. These observations suggest that the transition to G2 is not taking place as expected. In conclusion, Phasik detected the changes in temporal structure corresponding to the phenotypes described for the respective mutants.

Inferring cell cycle phases from gene expression data

High-quality ODE models as the one we used above[17] are so far only available for a few biological systems and are mostly restricted to systems with sufficient biological knowledge. Considering such models is very useful to validate new methods by showing that they can reproduce known results. However, it is also important to explore other types of data. We investigating whether we could also infer cell cycle phases from gene expression time series data. Such data are more easily available and provide temporal information for virtually all genes.

We used a time-series RNA-seq dataset described in Kelliher et al.[38] For this study, yeast cells had been first synchronized and then were released to undergo three full cell cycles, with RNA samples taken from the culture every 5 min for RNA-seq, providing RNA levels for all genes. One full cell cycle was reported to last 75 min. For our study, we only used the first cell cycle because the time series deteriorates as a result of de-synchronization of the cells. We downloaded normalized read counts and then normalized each time series so that values were between 0 and 1. As previously, we used the same static PPI network from KEGG, now deriving edge weights from gene expression data rather than protein concentrations of the nodes, to build a temporal network. Then, we inferred phases, following the workflow described above. The resulting temporal network contained 83 nodes and 159 edges, of which 158 contained temporal information over 16 time points.

Visual inspection of the distance matrix indicated a less clear temporal structure (Figure 6A) than that of the temporal network built from the ODE model. This could be due to a higher noise level in the data as well as less clear transitions in gene expression from one state to the next. When inferring clusters using Phasik (Figure 6B), we identified 4 stable clusters that roughly correspond to the 4 cell cycle phases and that persist at higher numbers of clusters. The average silhouette values were lower than the ones reported when using the temporal PPI derived from the ODE model, indicating a worse separation of clusters (Figure 6C).

Relating the resulting cell cycle phases from the temporal RNA-seq expression data to those inferred from the protein concentration data from the Chen et al.[17] model was challenging. We mainly used evidence of temporal profiles of cell cycle genes known to be temporally regulated in the cell cycle to assign cell cycle phases (taken from Koch and Nasmyth[7] and Nasmyth[7,9] with additional data from Lord et al.[39] and Amoussouvi et al.[39,40]) (Figure 6D). The phases we inferred from the clusters

CellPress
OPEN ACCESS

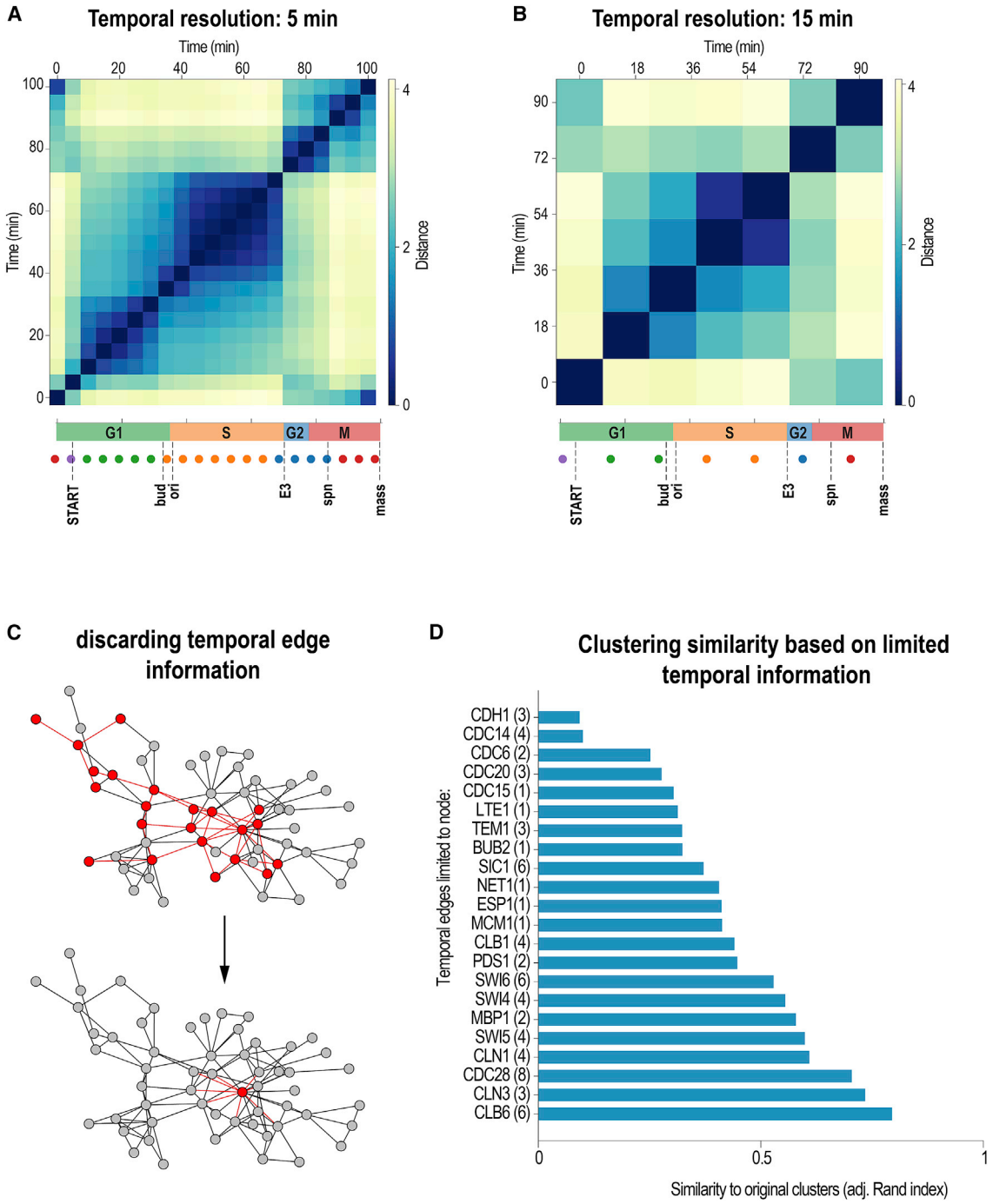


Figure 4. Discarding temporal information for phase recovery A limited temporal resultion and a few selected temporal edges were sufficient to recover the original 5 clusters

A limited temporal resultion and a few selected temporal edges were sufficient to recover the original 5 clusters.

(A and B) Biological phases inferred with down-sampled time series, using time points (A) every 5 min and (B) every 15 min. The original time step was 1 min. The main biological phases were still detected, with 15 min being the minimum time resolution needed to detect shorter phases.

(C) Schematic of discarding temporal edge information.

(D) Similarity of the newly computed 5 clusters to the original 5 clusters for each temporal node (numbers of temporal edges are given in parentheses). Values closer to 1 indicate clusters close to the original ones.

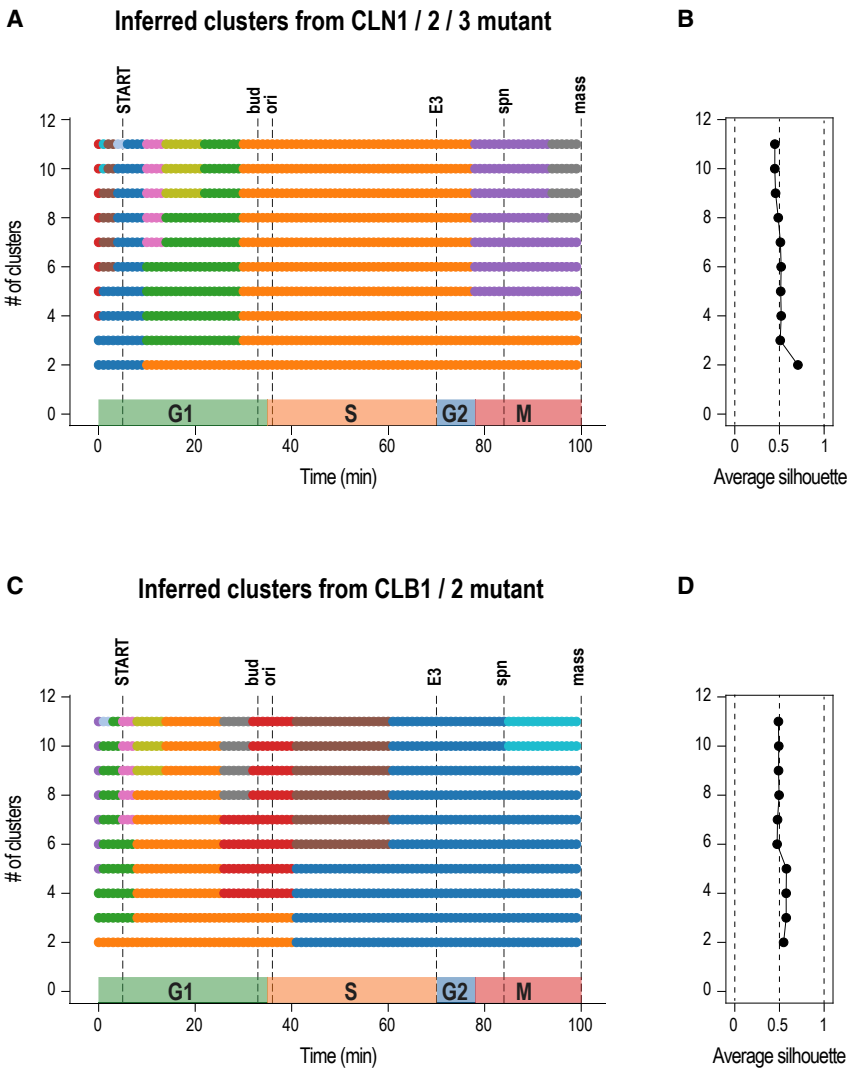


Figure 5. Inferring phases, together with the average silhouette, from cell cycle mutants for a range of timescales

(A–D) CLN1/2/3 knockout (A and B) and CLB1/2 knockout (C and D). The phase inference detected the expected G1 arrest in the CLN1/2/3 (A) and the G2 arrest in CLB1/2 (C). In (A), a late cluster (purple) appears because of amplification of quasi-constant time series by the normalization.

fitted reasonably well to the G1, S, G2, and M phases, especially for 4 and 5 clusters. Edge weights of the most active edges for 4 and 5 clusters are shown in Table S1. At higher numbers of clusters, we obtained new clusters that were similar to those obtained from the temporal network based on protein concentration data; namely, split of the M phase cluster with the pre-START G1 phase cluster, separation of S phase into an early and a later phase, separation of G1 phase into early and late, as well as split of M phase into two separate sub-phases.

Note that the biological entities are different when comparing Phasik results from RNA-seq (mRNA) data and the ODE model (protein); first, mRNA is produced prior to proteins, and second, both can underlie additional regulatory mechanisms influencing their temporal stability and degradation and, hence, their temporal expression patterns. It is therefore not unexpected that phases extracted by Phasik from these two different datatypes are different because phase inference heavily depends on the relative temporal expression strength of the nodes in the temporal network.

We also tested Phasik without adding the temporal network framework to see its requirement for phase inference from node time series. Including network information led to only minor improvements over the ODE time series alone (Figures S8A and S8B); however, clustering gene expression data alone yielded substantially worse clustering (Figures S8C and S8D). It seems therefore that making use of the static PPI network enhances the phase inference of experimental data such as RNA-seq time series, most likely by pre-selecting the subset of genes (nodes) in the network, together with their interactions.

Phasik identifies a phase shift in circadian mouse mutants

Circadian rhythms are another example of well-described, stable cellular oscillators. Based on a daily 24-h rhythm, life on earth has evolved mechanisms of adaption to daily light/dark cycles. The circadian clock, composed of key clock genes that regulate gene expression in an ∼24-h rhythm, is part of this adaptation mechanism[41]; in mammals, the transcriptional regulators Bmal1 (Arntl) and Clock drive expression of the clock genes *Per1*, *Per2*, *Cry1*, and *Cry2*. Their gene products Per and Cry form a complex and, when translocated to the nucleus, inhibit their own gene expression. This negative feedback loop leads to oscillatory gene expression dynamics of the Clock genes and their downstream targets that determine the daily rhythms of cellular physiology.

The mammalian circadian clock works at several levels, from the organism via the suprachiasmatic nucleus (SCN) down to tissue and the cell[42]; tissues show circadian oscillation in isolation, and almost all body cells contain a circadian clock.

Aviram et al.[34] investigated the circadian clock in isolated mouse liver in wild-type, *Per1/2*$^{-/-}$, and *Bmal*$^{-/-}$ mutants. They found a clock-independent ultradian rhythm of gene expression of ∼16 h in *Per1/2*$^{-/-}$ animals that was driven by the activity of the protein kinase Akt.

We applied Phasik to these RNA-seq datasets to identify the oscillatory behavior of a temporal network containing the genes of KEGG circadian rhythm (KEGG: mmu04710) and

CellPress
OPEN ACCESS

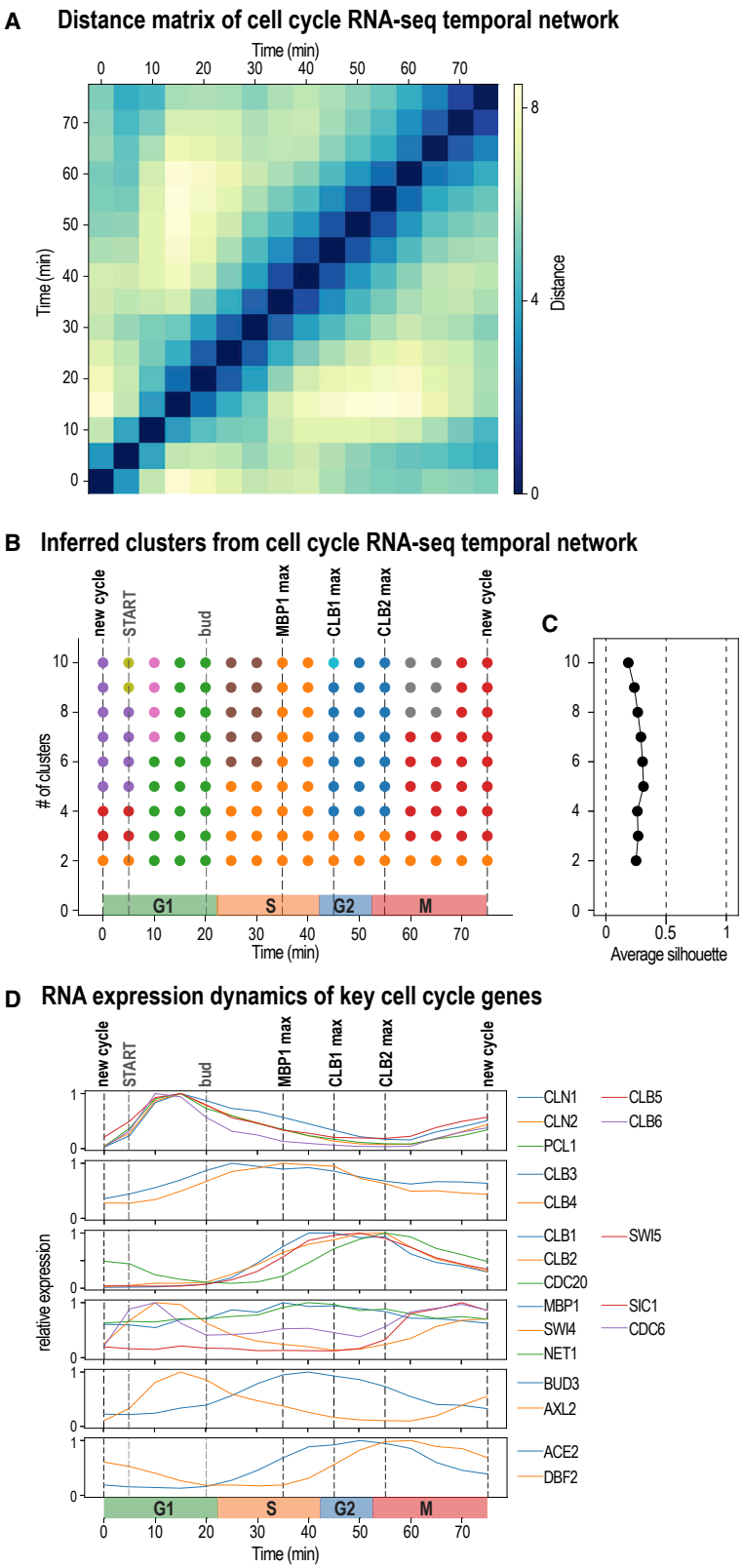


Figure 6. Phase inference from a temporal network constructed using RNA-seq data

(A) The distance matrix reflects the general coarser resolution of the data.

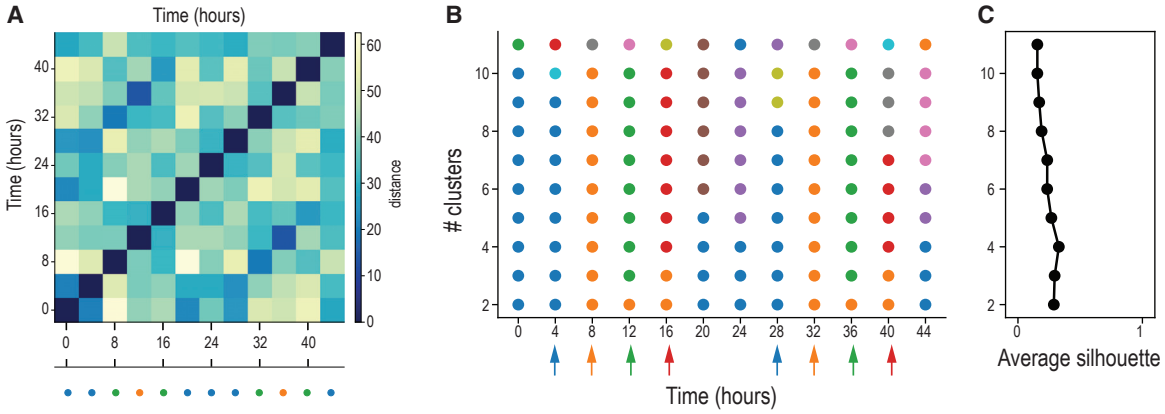
(B) The main 4 phases could be robustly inferred and were similar to those inferred from protein concentration data from the ODE model.

(C) Average silhouette values were low, indicating lower quality of the clustering.

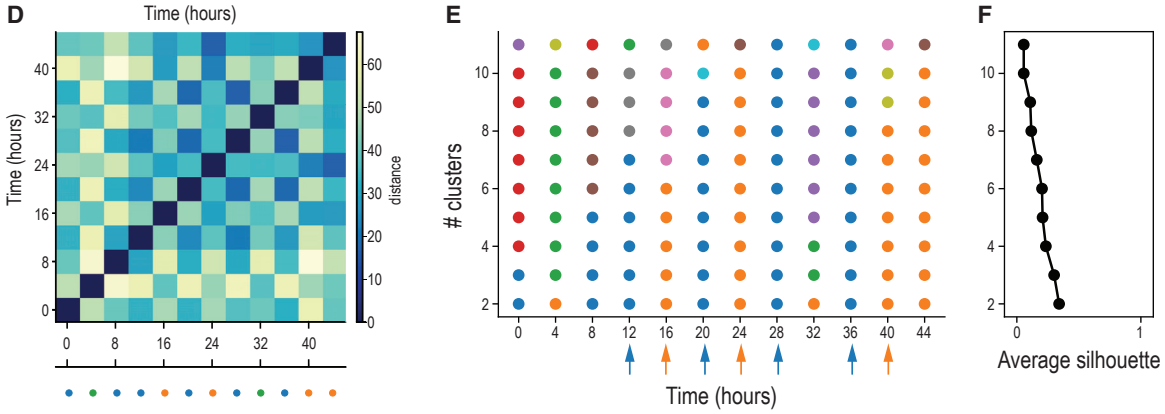
(D) Dynamics of RNA expression of key cell cycle genes.

mouse liver circadian rythm and entraining temporal network clustering

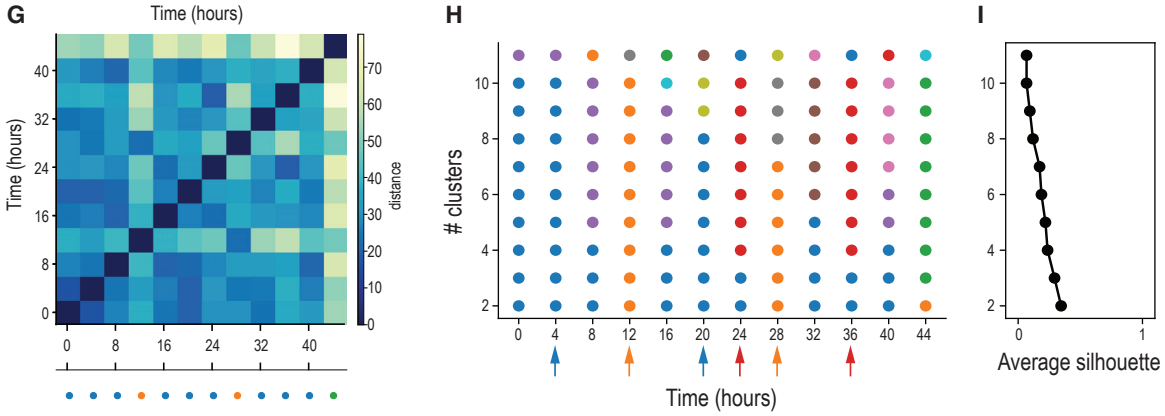
wild type



Per 1/2$^{-/-}$



Bmal$^{-/-}$



(legend on next page)

CellPress
OPEN ACCESS

circadian entrainment pathways (KEGG: mmu04713) and to observe changes induced upon perturbation of the system in *Per1/2*$^{-/-}$ and *Bmal*$^{-/-}$ mutants.

The wild-type mouse liver showed perfect oscillatory behavior of the network, with a 24-h rhythm that was apparent from 4 h onward as a block of 4 consecutive phases that was detectable up to high cluster numbers (Figures 7A–7C).

In *Per1/2*$^{-/-}$ livers, this rhythmic pattern was disturbed; there was a recurring pattern of 2 consecutive phases from 12 h onward, which repeated at 20 and 36 h and was detectable up to higher cluster numbers (Figures 7D–7F). Finally, the *Bmal*$^{-/-}$ liver also showed a weak oscillatory phase, repeating at 4 and 20 h, 12 and 28 h (both reflecting a 16-h rhythm), as well as one 12-h rhythm (24 to >36 h) (Figures 7G–7I).

Because Akt signaling has been described to be crucial for this ultradian rhythm, we also looked at the circadian behavior of the network of genes from the KEGG phosphatidylinositol 3-kinase (PI3K)/Akt signaling pathway (KEGG: mmu04151). In wild-type liver, the temporal network of PI3K-Akt signaling showed similar circadian patterns as the circadian network, with a clear 24-h rhythm of 4 consecutive phases (Figures S9A and S9B). In the *Per1/2*$^{-/-}$ liver, Phasik temporal network clustering uncovered at least two consecutive network states showing an ultradian rhythm, agreeing with results from Aviram et al.[34] (20 to >36 h, 24 to >40h; Figures S9C and S9D). Finally, the data from *Bmal*$^{-/-}$ liver mapped on the PI3K-Akt signaling pathway showed a mixed repetitive network pattern with at least one 16-h rhythm (12 to >28 h; Figures S9E and S9F). While we also saw a potential ultradian rhythm in the *Bmal*$^{-/-}$ knockout liver, the observed patterns were limited to one single phase. To summarize, Phasik analysis of circadian gene expression in the wild type could faithfully identify 24-h rhythmicity and confirmed an ultradian rhythm in *Per1/2*$^{-/-}$ as well as weakly in *Bmal*$^{-/-}$ mutant liver.

DISCUSSION

Here, we proposed a method to infer biological phases by representing temporal protein or gene expression data as a temporal network and by applying a clustering algorithm to the resulting series of snapshot networks. We validated our method by applying it to a well described biological system: the budding yeast cell cycle.

To build the temporal network, we started from a static PPI network in which we assigned time-evolving weights to certain edges, obtained by integrating time series of protein concentrations from an ODE model or from expression data from an RNA-seq study. We inferred biological phases from the temporal PPI network for a range of potential timescales by using a clustering algorithm with various numbers of clusters. While we used interaction data provided by KEGG to build the network and test our method, we obtained the same results when starting from a static network downloaded from the STRING database,[43] with the same 83 protein nodes as in KEGG; we tested one network with only physical interactions and one also including functional interactions (only high-confidence interactions from databases or experiments were used).

We tested our method against variations in chosen algorithms and parameters. We showed that our method is able to infer meaningful biological phases corresponding to the principal cell cycle phases G1, S, G2, and M, with a finer resolution in sub-phases at higher numbers of clusters. Identification of these cell cycle phases was possible with protein concentration data and gene expression data. We showed that our approach is robust against changes in distance metrics and clustering methods and identified those that performed best for our system of interest. We also showed that the method was general by detecting cell cycle phase arrests of mutants inducing G1 or G2 arrest.

We want to note here that, in addition to connectivity-based (hierarchical) and centroid-based (K-means) clustering methods, other common classes exist, such as density-based (e.g. DBSCAN [density-based spatial clustering of applications with noise] and OPTICS [ordering points to identify the clustering structure], good for outlier removal and clusters of arbitrary shapes) and distribution-based methods (e.g., Gaussian mixture models such as DPGMM (Dirichlet process Gaussian mixture models), good for incorporating covariance information of latent distributions).[44,45] We stress however, that, for our multiscale analysis, hierarchical clustering makes the most sense because it allows us to infer phases and sub-phases of processes and, hence, hierarchical temporal structures.

At each number of clusters, we were able to match the inferred phases to described biological phases and processes of the cell cycle. This may not be surprising because the mathematical model of Chen et al.[17] is based on extensive knowledge of key cell cycle events. It is remarkable, however, that we obtained these results even though our network is only partially temporal, with temporal information missing for nearly 80% of edges. We stress here that no additional information about cell cycle phases than the temporal network itself was used to infer these phases with Phasik.

We checked the contribution of the network framework in our analysis by clustering node time series directly to infer phases without using a network. Clustering ODE time series directly returned good clusters even without a network. This seems to be due to several specificities of the ODE data from this model because the most relevant proteins for the dynamics of the cell

Figure 7. Inferring oscillatory gene expression in the circadian rhythm in mouse wild-type, *Per1/2*$^{-/-}$, and *Bmal*$^{-/-}$ liver

(A and B) Distance matrix of the wild-type liver (A) and clusters of temporal network snapshots of the wild-type liver (B) showed a perfect 24-h rhythm up to high cluster numbers, indicated by colored arrows.

(C) The average silhouette is highest at 4 clusters.

(D and E) Distance matrix of the *Per1/2*$^{-/-}$ liver (D) and clusters of temporal network snapshots of the *Per1/2*$^{-/-}$ liver (E) showed 16-h periodicity as well as an overlaid 8-h rhythm.

(F) The average silhouette score for (E).

(G and H) Distance matrix of the *Bmal*$^{-/-}$ liver (G); 16-h rhythmic expression dynamics are also detectable in the *Bmal*$^{-/-}$ liver (H).

(I) The average silhouette score for (H).

cycle and whose time evolution was most informative for building the model had been carefully selected.[17] This selection is similar to use of a network on other datasets, such as RNA-seq data. Also, the ODE data are virtually noise free. Finally, the ODE time series include a few time series that actually represent edges corresponding to some of the most central protein interactions in the cell cycle (and, hence, network elements) and not nodes. All of these facts make ODE time series data ideal to detect phases directly without using a network.

Clustering gene expression without network information returned substantially worse clusters. Thus, for unbiased datasets with a higher noise level, using a network seems to enhance phase inference by selecting a smaller subset of genes from the temporal dataset and the subset of biologically relevant edges between them. This suggests that using a network is especially useful when using primary experimental time series data by adding pre-existing biological knowledge in form of a PPI. For well-curated, processed temporal data such as the ODE model of Chen et al.,[17] which already contain some information on interacting components, providing a network in the form of a PPI in addition seems to be partially dispensable because this leads only to minor improvements over the ODE data alone. Our method is intended to be used primarily on less well-known systems and using unprocessed and noisy data for which the network framework clearly adds valuable information. It should be noted that, should a network structure not be available, the network could also be built from other sources, such as co-expression of genes. Here, it would be interesting to infer the underlying network directly from time series data of an oscillatory system, as described here.[46]

We applied Phasik to a second, highly rhythmic system: the circadian clock. We used only RNA-seq data of mouse liver collected every 4 h over a period of 2 days from wild-type, *Per1/2*$^{-/-}$, and *Bmal*$^{-/-}$ animals. Under wild-type conditions, we observed clear 24-h circadian rhythm of 2 cellular networks: circadian rhythm and entrainment and PI3K-Akt signaling. Ultradian, 16-h phases were observable in *Per1/2*$^{-/-}$ liver. This agrees with the observations described in Aviram et al.,[34] although Phasik also identified a weak ultradian rhythm in the *Bmal*$^{-/-}$ liver.

A peculiarity of the systems we considered is that only partial temporal information was available. Usual temporal network studies consider all edges to include temporal information (either discretized or with evolving weights). Here, we further investigated this point and the specific role of the temporal information in inferring biologically meaningful phases. First, we showed that the inferred clusters remained almost unchanged when we discarded the temporal information of any single edge. Second, we showed that meaningful clusters could still be obtained even when using temporal information for the edges of some single nodes. The results depended on the node selected, suggesting that some nodes carry more important temporal information than others. For example, MBP1 (2 temporal edges) and CDC28 (8 temporal edges) yielded results comparable with the ones obtained with the original network but CDC14 (2 temporal edges) did not. Interestingly, we found no correlation between the number of temporal edges of a node and its ability to recover the 5 clusters of the cell cycle. Furthermore, no correlation with typical static centrality measures could be observed, including degree and betweenness centrality, and we observed only weak correlation with eigenvector centrality. Third, we showed that the minimum sampling frequency was dictated by the timescales of the phases to be detected but that sampling frequency had minor effects on clusters at values above that threshold. Many interesting open questions remain for theoreticians and experimentalists concerning partial temporal networks. By understanding which interactions are most informative about the temporal structure of a system, experimentalists could focus primarily on those. For instance, while it is well known that CDC28 is a major driver of cell cycle events, major drivers might be less well known in other biological systems. Yet, our method shows that obtaining even partial temporal information on some components might be sufficient to gain insights into the temporal structure of a (biological) process.

To test the Phasik workflow, we used a well-defined time series of protein concentrations obtained from an ODE model. These data have a virtually infinite sampling frequency and are noise free. Moreover, such models typically contain just a handful of carefully selected proteins to reproduce the dynamics of a specific biological system. Such complete and high-quality models are not available for many biological systems because they require a large body of experimental evidence and modeling effort. More accessible temporal data can be obtained by time-resolved high-throughput measurements of RNA or protein levels. Such studies have become standard and are technically easier. However, high-throughput measurements come with downsides: inherent high noise levels; a coarse temporal resolution, typically in the range of hours rather than minutes; and, in the case of RNA-seq, transitions in RNA expression levels from one state to the next that are much less distinctive than for proteins. While the temporal RNA-seq data from the yeast cell cycle suffered from all of these issues, we succeeded in inferring the 4 relevant cell cycle phases when using them. Thus, while ODE models have many advantages over temporal expression data for the present application, the latter is a more available option for biological systems in general and can be readily analyzed using the Phasik workflow.

In conclusion, we proposed a method, Phasik, to infer phases of biological systems using temporal networks. We inferred phases and sub-phases of biological systems by modeling them as a partially temporal networks of PPIs by combining static networks with protein concentration or gene expression data and then clustering the temporal network snapshots. We systematically tested the robustness of the results against variations in the algorithm and the data used. We showed the general applicability of Phasik by using it for phase inference on the budding yeast cell cycle as well as on circadian data from the mouse. We made our code publicly available for use on other biological systems of interest (https://gitlab.com/habermann_lab/phasik).

Limitations of the study

Phasik has several limitations. First, the quality of the results obtained with Phasik depends on the availability of high-quality and high-resolution temporal expression data (gene or protein or a mathematical model) as well as suitable PPI data to build

CellPress
OPEN ACCESS

a (partially) temporal network of interest for a specific process. For many biological systems, these data are currently not available. Second, we optimized the clustering algorithm and distance metrics used to get the best clustering for our systems: cell cycle and circadian rhythm. It should be noted that clustering algorithms could perform differently for other types of data. Carefully evaluating different methods would therefore be recommended for other systems. Third, if a system has only weak inherent temporal structure, then the interpretation of the results might be difficult. Finally, predicted phases should be carefully interpreted and experimentally validated; e.g., by disturbing the system through interfering with the expression or function of proteins important for the temporal behavior. The most important protein interactions for each phase are returned by Phasik and would be good candidates for further experimental studies.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Building the temporal network: Integrating temporal information to a static network
  - Inferring biological phases from a temporal PPI network by clustering snapshots
  - Measuring clustering quality
  - Building of static PPI network of the cell cycle from KEGG
  - Times series data for edge activities
  - Protein concentrations
  - Gene expression from RNA-sequencing data of the cell cycle
  - Temporal networks of PPI of the cell cycle
  - Building temporal networks for mouse circadian data
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2023.100397.

ACKNOWLEDGMENTS

We thank Béla Novàk for early discussions of their model. We thank the members of the Habermann team for helpful discussions and critical reading of the manuscript. The project leading to this publication has received funding from France 2030, the French Government program managed by the French National Research Agency (ANR-16-CONV-0001) and from Excellence Initiative of Aix-Marseille University - A*MIDEX. A.B. acknowledges partial support from the ANR project DATAREDUX (ANR-19-CE46-0008-01).

AUTHOR CONTRIBUTIONS

M.L., B.H., L.T., and A.B. conceived the study. M.L. was mainly responsible for code implementation and data analysis with help from A.M. and A.T.-T. M.L., B.H., L.T., and A.B. were responsible for data interpretation. M.L. and B.H. wrote the manuscript with input from L.T., A.B., A.M., and A.T.-T.

DECLARATION OF INTERESTS

The authors declare no competing interests.

INCLUSION AND DIVERSITY

We support inclusive, diverse, and equitable conduct of research.

Received: July 31, 2022
Revised: August 13, 2022
Accepted: January 11, 2023
Published: February 1, 2023

REFERENCES

1. Buijs, R.M., van Eden, C.G., Goncharuk, V.D., and Kalsbeek, A. (2003). The biological clock tunes the organs of the body: timing by hormones and the autonomic nervous system. J. Endocrinol. *177*, 17–26. https://doi.org/10.1677/joe.0.1770017.
2. Goldbeter, A., and Berridge, M.J. (1996). Biochemical Oscillations and Cellular Rhythms: The Molecular Bases of Periodic and Chaotic Behaviour, 1st edition (Cambridge University Press). https://doi.org/10.1017/CBO9780511608193.
3. Refinetti, R., and Menaker, M. (1992). The circadian rhythm of body temperature. Physiol. Behav. *51*, 613–637. https://doi.org/10.1016/0031-9384(92)90188-8.
4. Laurenti, E., and Göttgens, B. (2018). From haematopoietic stem cells to complex differentiation landscapes. Nature *553*, 418–426. https://doi.org/10.1038/nature25022.
5. Loomis, A.L., Harvey, E.N., and Hobart, G.A. (1937). Cerebral states during sleep, as studied by human brain potentials. J. Exp. Psychol. *21*, 127–144. https://doi.org/10.1037/h0057431.
6. Howard, A., and Pelc, S.R. (1953). Synthesis of deoxyribonucleic acid in normal and irradiated cells and its relation to chromosome breakage. Heredity *6*, 261–273.
7. Koch, C., and Nasmyth, K. (1994). Cell cycle regulated transcription in yeast. Curr. Opin. Cell Biol. *6*, 451–459. https://doi.org/10.1016/0955-0674(94)90039-6.
8. Murray, A.W. (2004). Recycling the cell cycle: cyclins revisited. Cell *116*, 221–234. https://doi.org/10.1016/S0092-8674(03)01080-8.
9. Nasmyth, K. (1993). Control of the yeast cell cycle by the Cdc28 protein kinase. Curr. Opin. Cell Biol. *5*, 166–179. https://doi.org/10.1016/0955-0674(93)90099-C.
10. Nasmyth, K. (1996). At the heart of the budding yeast cell cycle. Trends Genet. *12*, 405–412. https://doi.org/10.1016/0168-9525(96)10041-X.
11. Vodermaier, H.C. (2004). APC/C and SCF: controlling each other and the cell cycle. Curr. Biol. *14*, R787–R796. https://doi.org/10.1016/j.cub.2004.09.020.
12. Newman, M. (2018). Networks (Oxford University Press).
13. Alon, U. (2003). Biological networks: the tinkerer as an engineer. Science *301*, 1866–1867. https://doi.org/10.1126/science.1089072.
14. Pavlopoulos, G.A., Secrier, M., Moschopoulos, C.N., Soldatos, T.G., Kossida, S., Aerts, J., Schneider, R., and Bagos, P.G. (2011). Using graph theory to analyze biological networks. BioData Min. *4*, 10. https://doi.org/10.1186/1756-0381-4-10.

15. Kanehisa, M., and Goto, S. (2000). KEGG: kyoto encyclopedia of genes and genomes. Nucleic Acids Res. *28*, 27–30. https://doi.org/10.1093/nar/28.1.27.

16. Chen, T., He, H.L., and Church, G.M. (1998). Modeling gene expression with differential equations. In Biocomputing '99, pp. 29–40.

17. Chen, K.C., Calzone, L., Csikasz-Nagy, A., Cross, F.R., Novak, B., and Tyson, J.J. (2004). Integrative analysis of cell cycle control in budding yeast. Mol. Biol. Cell *15*, 3841–3862. https://doi.org/10.1091/mbc.e03-11-0794.

18. Traynard, P., Fauré, A., Fages, F., and Thieffry, D. (2016). Logical model specification aided by model-checking techniques: application to the mammalian cell cycle regulation. Bioinformatics *32*, i772–i780. https://doi.org/10.1093/bioinformatics/btw457.

19. Holme, P. (2015). Modern temporal network theory: a colloquium. Eur. Phys. J. B *88*, 234. https://doi.org/10.1140/epjb/e2015-60657-4.

20. Holme, P., and Saramäki, J. (2012). Temporal networks. Phys. Rep. *519*, 97–125. https://doi.org/10.1016/j.physrep.2012.03.001.

21. Gelardi, V., Godard, J., Paleressompoulle, D., Claidiere, N., and Barrat, A. (2020). Measuring social networks in primates: wearable sensors versus direct observations. Proc. Math. Phys. Eng. Sci. *476*, 20190737. https://doi.org/10.1098/rspa.2019.0737.

22. Miritello, G., Moro, E., and Lara, R. (2011). Dynamical strength of social ties in information spreading. Phys. Rev. E Stat. Nonlin. Soft Matter Phys. *83*, 045102. https://doi.org/10.1103/PhysRevE.83.045102.

23. Saramäki, J., and Moro, E. (2015). From seconds to months: an overview of multi-scale dynamics of mobile telephone calls. Eur. Phys. J. B *88*, 164. https://doi.org/10.1140/epjb/e2015-60106-6.

24. Lopes, M.A., Zhang, J., Krzemiński, D., Hamandi, K., Chen, Q., Livi, L., and Masuda, N. (2021). Recurrence quantification analysis of dynamic brain networks. Eur. J. Neurosci. *53*, 1040–1059. https://doi.org/10.1111/ejn.14960.

25. Pedreschi, N., Bernard, C., Clawson, W., Quilichini, P., Barrat, A., and Battaglia, D. (2020). Dynamic core-periphery structure of information sharing networks in entorhinal cortex and hippocampus. Netw. Neurosci. *4*, 946–975. https://doi.org/10.1162/netn_a_00142.

26. Przytycka, T.M., Singh, M., and Slonim, D.K. (2010). Toward the dynamic interactome: it's about time. Brief. Bioinform. *11*, 15–29. https://doi.org/10.1093/bib/bbp057.

27. Li, M., Yang, J., Wu, F.-X., Pan, Y., and Wang, J. (2018). DyNetViewer: a Cytoscape app for dynamic network construction, analysis and visualization. Bioinformatics *34*, 1597–1599. https://doi.org/10.1093/bioinformatics/btx821.

28. Pierrelée, M., Reynders, A., Lopez, F., Moqrich, A., Tichit, L., and Habermann, B. (2020). TimeNexus: a novel cytoscape app to analyze time-series data using temporal MultiLayer networks (tMLNs). Preprint at Researchsquare. https://doi.org/10.21203/rs.3.rs-133258/v1.

29. Chechik, G., Oh, E., Rando, O., Weissman, J., Regev, A., and Koller, D. (2008). Activity motifs reveal principles of timing in transcriptional control of the yeast metabolic network. Nat. Biotechnol. *26*, 1251–1259. https://doi.org/10.1038/nbt.1499.

30. Komurov, K., and White, M. (2007). Revealing static and dynamic modular architecture of the eukaryotic protein interaction network. Mol. Syst. Biol. *3*, 110. https://doi.org/10.1038/msb4100149.

31. Li, M., Wu, X., Wang, J., and Pan, Y. (2012). Towards the identification of protein complexes and functional modules by integrating PPI network and gene expression data. BMC Bioinf. *13*, 109. https://doi.org/10.1186/1471-2105-13-109.

32. Ou-Yang, L., Dai, D.-Q., Li, X.-L., Wu, M., Zhang, X.-F., and Yang, P. (2014). Detecting temporal protein complexes from dynamic protein-protein interaction networks. BMC Bioinf. *15*, 335. https://doi.org/10.1186/1471-2105-15-335.

33. Masuda, N., and Holme, P. (2019). Detecting sequences of system states in temporal networks. Sci. Rep. *9*, 795. https://doi.org/10.1038/s41598-018-37534-2.

34. Aviram, R., Dandavate, V., Manella, G., Golik, M., and Asher, G. (2021). Ultradian rhythms of AKT phosphorylation and gene expression emerge in the absence of the circadian clock components Per1 and Per2. PLoS Biol. *19*, e3001492. https://doi.org/10.1371/journal.pbio.3001492.

35. Wallach, T., Schellenberg, K., Maier, B., Kalathur, R.K.R., Porras, P., Wanker, E.E., Futschik, M.E., and Kramer, A. (2013). Dynamic circadian protein–protein interaction networks predict temporal organization of cellular functions. PLoS Genet. *9*, e1003398. https://doi.org/10.1371/journal.pgen.1003398.

36. Rousseeuw, P.J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. *20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7.

37. Lovrics, A., Csikász-Nagy, A., Zsély, I.G., Zádor, J., Turányi, T., and Novák, B. (2006). Time scale and dimension analysis of a budding yeast cell cycle model. BMC Bioinf. *7*, 494. https://doi.org/10.1186/1471-2105-7-494.

38. Kelliher, C.M., Leman, A.R., Sierra, C.S., and Haase, S.B. (2016). Investigating conservation of the cell-cycle-regulated transcriptional program in the fungal pathogen, cryptococcus neoformans. PLoS Genet. *12*, e1006453. https://doi.org/10.1371/journal.pgen.1006453.

39. Lord, M., Yang, M.C., Mischke, M., and Chant, J. (2000). Cell cycle programs of gene expression control morphogenetic protein localization. J. Cell Biol. *151*, 1501–1512. https://doi.org/10.1083/jcb.151.7.1501.

40. Amoussouvi, A., Teufel, L., Reis, M., Seeger, M., Schlichting, J.K., Schreiber, G., Herrmann, A., and Klipp, E. (2018). Transcriptional timing and noise of yeast cell cycle regulators—a single cell and single molecule approach. NPJ Syst. Biol. Appl. *4*, 17–10. https://doi.org/10.1038/s41540-018-0053-4.

41. Cox, K.H., and Takahashi, J.S. (2019). Circadian clock genes and the transcriptional architecture of the clock mechanism. J. Mol. Endocrinol. *63*, R93–R102. https://doi.org/10.1530/JME-19-0153.

42. Mohawk, J.A., Green, C.B., and Takahashi, J.S. (2012). Central and peripheral circadian clocks in mammals. Annu. Rev. Neurosci. *35*, 445–462. https://doi.org/10.1146/annurev-neuro-060909-153128.

43. Jensen, L.J., Kuhn, M., Stark, M., Chaffron, S., Creevey, C., Muller, J., Doerks, T., Julien, P., Roth, A., Simonovic, M., et al. (2009). STRING 8--a global view on proteins and their functional interactions in 630 organisms. Nucleic Acids Res. *37*, D412–D416. https://doi.org/10.1093/nar/gkn760.

44. Müllner, D. (2011). Modern hierarchical, agglomerative clustering algorithms. Preprint at arXiv. https://doi.org/10.48550/arXiv.1109.2378.

45. Jain, A.K. (2010). Data clustering: 50 years beyond K-means. Pattern Recognit. Lett. *31*, 651–666. https://doi.org/10.1016/j.patrec.2009.09.011.

46. Tyler, J., Forger, D., and Kim, J. (2021). Inferring causality in biological oscillators. Bioinformatics *38*, 196–203. https://doi.org/10.1093/bioinformatics/btab623.

47. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2.

48. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: machine learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.

CellPress
OPEN ACCESS

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
S. cerevisiae static cell cycle network	KEGG: sce04111	https://doi.org/10.1093/nar/28.1.27
ODE model of *S. cerevisiae* cell cycle	ODE model @ mpf.biol.vt.edu	https://doi.org/10.1091/mbc.e03-11-0794
temporal RNA-sequencing of *S. cerevisiae* cell cycle	GEO: GSE80474	https://doi.org/10.1371/journal.pgen.1006453
M. musculus static network of circadian rhythm	KEGG: mmu04710	https://doi.org/10.1093/nar/28.1.27
M. musculus static network of circadian rhythm	KEGG: mmu04713	https://doi.org/10.1093/nar/28.1.27
M. musculus network of Pi3K/Akt signaling	KEGG: mmu04151	https://doi.org/10.1093/nar/28.1.27
temporal RNA-sequencing dataset of *M. musculus* circadian rhythm	GEO: GSE171975	https://doi.org/10.1371/journal.pbio.3001492
Software and algorithms		
Phasik	https://gitlab.com/habermann_lab/phasik	https://doi.org/10.5281/zenodo.7378779

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Bianca Habermann (bianca.HABERMANN@univ-amu.fr).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- This paper analyses existing, publicly available data. These accession numbers for the datasets are listed in the key resources table.
- All original code has been deposited at https://gitlab.com/habermann_lab/phasik and is publicly available. DOIs are listed in the key resources table.
- Any additional information required to re-analyse the data reported in this paper is available from the lead contact upon request.

METHOD DETAILS

Building the temporal network: Integrating temporal information to a static network

A *temporal network* is a network where the edges connecting the nodes can vary over time. In a weighted network, this means that the edge weights are time-varying. We build such a (weighted) temporal network by incorporating time series of edge weights, which represent the activity of the corresponding interactions, into a static network. Note that we only consider undirected interactions.

In the case of proteins, actual protein-protein interactions are difficult to measure over time. Quantities relative to each protein are more accessible however, such as protein concentrations or associated gene expression levels. We used these (node) time series to generate corresponding edge time series. We multiplied the time series relative to protein A by that of protein B to obtain a time series for the interaction A-B. We did so only if this edge A-B exists in the static network. It is also possible that no edge time series for an edge A-B in the static network exists: this can happen if there are no time series for proteins A or B. In these cases, we set the weight to a constant value.

Our code to build temporal networks as described is available online (https://gitlab.com/habermann_lab/phasik). Several utility functions are available to easily integrate node or edge times series, e.g. RNA-seq data, to a static network. Details about the actual data used are provided below.

Inferring biological phases from a temporal PPI network by clustering snapshots

A temporal network can be seen as a list of snapshots, where each snapshot is the adjacency matrix of the network at a given time. We expect two snapshots S and S' at times t and t' to be similar if the system is in the same state or phase at those times. On the

contrary, if times *t* and *t'* correspond to different phases, we expect the snapshots at those times to be very different. The underlying assumption is that the structure of the network is linked to the state that the system is in.

This idea can be formalised by clustering the snapshots of a given temporal network (Masuda and Holme 2019).[33] Each snapshot is a data point and is assigned to a cluster. As an output, we obtain clusters composed of snapshots: if snapshots *S* and *S′* are in the same cluster, it means that the system is in the same phase at times *t* and *t'*. We mainly used hierarchical clustering and *k*-means clustering. These were implemented with functions from Scipy[47] and Scikit-learn,[48] respectively. To compute the distance between snapshots, we flattened each adjacency matrix and then used Euclidean distance as a vector distance metric, unless stated otherwise. To compute the distance between clusters in the hierarchical clustering, we used the Ward variance minimisation method, unless stated otherwise. The desired number of clusters was set *a priori*, both in the hierarchical clustering used and in *k*-means.

Measuring clustering quality

To check the quality of computed clusters, we used the *silhouette score*,[36] which ranges from −1 to 1. For a given data point, the silhouette score is larger if it is close to other data points in the same cluster (cohesion) but far from data points in other clusters (separation). The *average silhouette score* is obtained by averaging over all data points and is a measure of how well separated the computed clusters are. Low scores indicate clusters of bad quality. To compare clusterings, we used the adjusted Rand index which ranges from 0 to 1. Its value is 1 when the two sets of clusters are identical, but close to 0 for random cluster assignments.

Building of static PPI network of the cell cycle from KEGG

We built a static PPI network of the budding yeast cycle from the manually curated regulatory network downloaded from the KEGG database[15] (KEGG: sce04111). We built our PPI in two main steps: we first merged duplicate nodes and then converted nodes and edges of multiple types to nodes of a single type connected by undirected edges of a single type. The final PPI network, shown in Figure 2A, consists of 83 proteins (nodes) and 159 protein-protein interactions (edges). Note that both protein concentration and RNA-seq expression data can be mapped to this network. The nodes thus represent both proteins and genes.

In the KEGG network, nodes can be of 5 types: gene, group, compound, map, and ortholog. First, we discarded the 2 compound, 4 map, and 3 ortholog nodes which do not represent genes or proteins, and kept only the 125 gene nodes (e.g. CDC28) and the 27 group nodes (e.g. [CDC28, CLN3], representing sets of gene nodes). We then merged duplicate nodes and removed disconnected nodes. For group nodes, we distinguished *true* groups from *normal* groups: true groups are node groups with components only interacting among themselves except for one edge that connects them to external nodes. This distinction allowed us to treat true groups as one node (e.g. nodes [SMC1, SMC3, MCD1, IRR1] in the PPI network were replaced by a new node, Cohesin). This resulted in 83 nodes in the static PPI network. Second, we converted the 88 KEGG relations to edges in the PPI network. KEGG relations are of 3 types: gene-gene, gene-group, and group-group. For gene-gene relations, we simply added an edge between the corresponding nodes in the PPI network. For relations involving groups, we treated them as gene-gene in case of true groups. In the case of normal groups, we added an edge between each node of the group and the external node or nodes. Finally, we added edges between each pair of nodes that are part of the same normal group, and removed self-edges in the PPI network. This resulted in the 159 edges in the PPI network.

Times series data for edge activities

We used two different datasets and built two different temporal networks by integrating them to the static PPI network presented above.

Protein concentrations

We used time series of protein concentrations obtained from the reference ODE model of the budding yeast cell cycle.[17] This mathematical model, based on and validated by extensive experimental data, describes the evolution of the concentration of a selected number of proteins over the entire cell cycle, based on knowledge about their interactions. We obtained the time series of protein concentrations by simulating the ODE model from the publicly available script of[17] with a time step of 1 min. The cell cycle in that study lasts 101 min, and hence there are 101 timepoints for one full cycle. The model consists of 46 variables that can be divided into 3 types: proteins (e.g. CDC6), protein complexes (e.g. the variable C2 represents the complex CLB1-SIC1), and 4 special variables (MASS, BUD, ORI, SPN) that serve as indicators of specific events along the cycle. Note that some proteins in the static PPI network are not represented in the ODE model, and vice versa. In addition, similar cyclins are represented by a single variable in the model, hence, we used CLN1, CLB1, CLB3, and CLB6 to represent CLN1/2, CLB1/2, CLB3/4, and CLB5/6, respectively.

For proteins *A* and *B* in the model, we defined the *activity* $w_{A\text{-}B}(t)$ of their interaction at time *t* as the product of their respective concentrations: $w_{A\text{-}B}(t) = [A](t) \times [B](t)$, if there is an edge *A-B* in the static PPI network. Some edges in the PPI network are represented by a single "protein complex" variable in the model: in those cases, we defined the activity of that edge as the concentration of that single variable. We then normalised each of these times series.

To simulate the mutants CLN1/$^2/_3$ and CLB1/2, we used the same ODE model as above, and only changed the values of a few key parameters, following Chen et al. Specifically, we set ksn2' = ksn2" = 0, Dn3 = 0 for CLN1/2/3 and ksb2' = ksb2" = 0 for CLB1/2.

CellPress
OPEN ACCESS

Gene expression from RNA-sequencing data of the cell cycle

To create a temporal cell cycle network based on RNA-sequencing data, we used mRNA counts of a gene expression time series, normalised to the library size, from.[38] We only considered the 15 timepoints of the first full cell cycle, starting at minute 10 in the dataset.

We defined the activity of a protein interaction *A-B* over time by the product of the respective RNA counts of the corresponding genes, as above: $w_{A\text{-}B}(t) = N_A(t) \times N_B(t)$. We then normalised each of these times series.

Temporal networks of PPI of the cell cycle

We built two (partially) temporal networks of protein interactions for one full cell cycle of budding yeast: for both, we used the static PPI network described above, to which we integrated the edge activities from (i) protein concentrations, and (ii) the gene expression data from RNA-sequencing. Both contained 83 nodes and 159 edges, but a different number of *temporal* edges. In both cases, we set the weight of the other edges that lack temporal information to a constant value of 1.

Building temporal networks for mouse circadian data

We used data from[34] (GEO: GSE171975). We downloaded raw read counts, normalized for library sizes and averaged data for each time point over all replicates. The circadian network was assembled by downloading all genes from the mouse Circadian Rhythm (KEGG: mmu04710) and Circadian Entrainment (KEGG: mmu04713) networks from the KEGG database, for PI3K-Akt signaling, we used network (KEGG: mmu04151). Protein-protein interactions connecting network components were in turn downloaded from the STRING database to create the circadian network, considering only high-confidence interactions from experiment and database.

QUANTIFICATION AND STATISTICAL ANALYSIS

To assess the quality of a given clustering, we used the average silhouette score (averaged over all data points) as well as the sample silhouette score for each data point. To compare the similarity between two clusterings, we used the adjusted Rand index. To quantify the correlation between the adjusted Rand index and network centrality measures, we used the R-squared value. For more details, see "Measuring clustering quality" in STAR Methods.