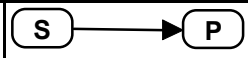
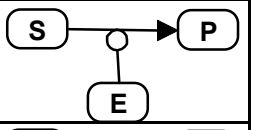
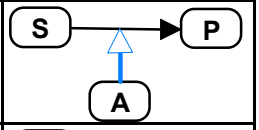
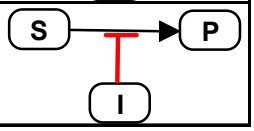
Supplementary data

1. Detailed description of regulator-reaction equations

Supplementary Table 1 Regulator-reaction equation and its diagram notation

	Regulator Reaction	Chemical Reaction	Graphical Notation
Mechanistic Model	S→P	S→P	
	E -o S→P E: Enzyme	E+S <-> E:S →P	
Semantic Model	A ->> S→P A: Activator		
	I –|| S→P I: Inhibitor		

Chemical reaction equations are able to describe mechanistic pathways of metabolic and gene regulatory networks, but they are neither suitable for describing unknown pathways nor do they represent a meaningful flow of a signal transduction pathway. They do not indicate the architecture of regulation, since they do not distinguish regulators (such as enzymes, activators, and inhibitors) explicitly from their associated regulated processes. In order to overcome these problems, the regulator-reaction equations (RREs) have been proposed, which describe not only mechanistic chemical reactions, but also a semantic model of a meaningful flow of pathways by specifying regulators and their regulated processes. The fundamental notation of the RRE is provided by:

Regulator Switch (-o /->>/-ll) Substrates -> Products,

where a reaction process is divided into the regulator part and the associated regulated process. The regulator is classified as one of three different categories: enzyme (-o), activator (->>), or inhibitor (-ll). When an enzyme is regarded as a regulator, it catalyzes the molecular reaction indicated by the arrow. In activation or inhibition processes, the regulator symbol acts on the molecular reaction. Activator- or inhibitor- type regulators can regulate a semantic process that shows the meaningful flow of pathways without knowing any detailed mechanism, such as a shortcut for a complicated regulation or gene expression. On the other hand, the mechanistic model indicates the physical reality of molecular interactions such as enzyme-substrate complex formation and binding between molecules. The RRE represents both mechanistic and semantic models explicitly.

Supplementary Table 2 A rule of how to determine regulation status

Regulation status is determined by:
(Regulation status)
= (Reaction type?)×(Which part is a key found?)×(Which part does a second node come from?)
or

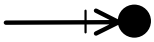
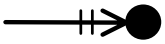
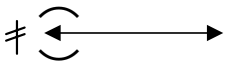
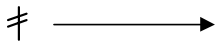
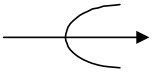
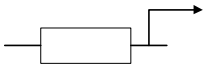
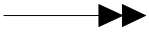
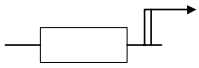
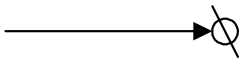
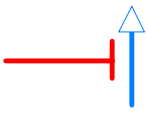
$$(regulation\ status) = \begin{pmatrix} 1(no\ regulator) \\ 1(enzyme) \\ 1(activation) \\ -1(suppression) \end{pmatrix} \begin{pmatrix} 1(regulator\ part) \\ 1(substrate\ part) \\ -1(product\ part) \end{pmatrix} \begin{pmatrix} -1(substrate\ part) \\ 1(product\ part) \end{pmatrix}.$$

Details of the reaction type are summarized in **Supplementary Table 3**. For example, if a key and its second messenger species are found in the regulator and substrate parts of the enzyme reaction, respectively, their regulation status is provided by: *1* (enzyme) × *1* (regulator part) × *-1*(substrate part) = *-1* (suppression).

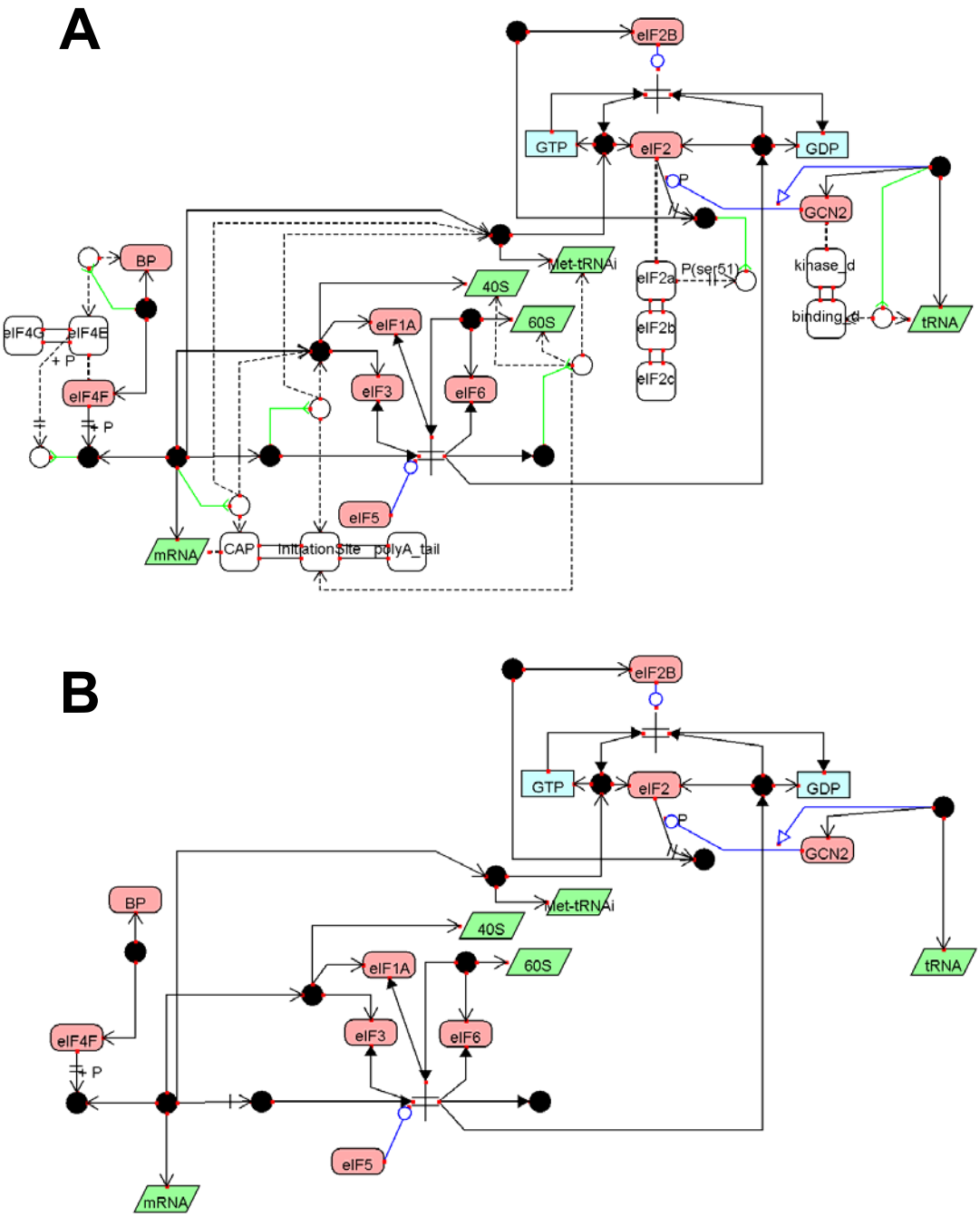
Regulator Type	ReactionType	Which part is a key found?	Which part does a second messenger come from?
	binding homo association or modification	substrate	substrate/product
		product	substrate
enzyme activator inhibitor	binding with stoichiometric changes	regulator/substrate	substrate/product
enzyme activator inhibitor	homo association or modification with stoichiometric changes	regulator/substrate	substrate/product
	elimination	substrate	product
enzyme activator inhibitor	elimination with stoichiometric changes	regulator/substrate	substrate/product
	reversible conversion	substrate	substrate/product
		product	substrate/product
enzyme activator inhibitor	irreversible conversion	regulator/substrate	substrate/product
	reversible conversion regarding multicomponents	substrate	substrate/product
		product	substrate/product
enzyme activator inhibitor	irreversible conversion regarding multicomponents	regulator/substrate	substrate/product
enzyme activator inhibitor	transport	regulator/substrate	product
enzyme activator inhibitor	degradation	regulator/substrate	substrate
inhibitor activator	transcription translation protein synthesis	regulator/substrate	product
	activation of species	substrate	product
	inhibition of species	substrate	product
activator inhibitor	activation modifier Inhibitor –>> #ID	regulator	product of #ID regulator of #ID
	inhibition modifier Inhibitor –II #ID	regulator	product of #ID regulator of #ID

#ID indicates an RRE.

Supplementary Table 3 Reaction Type defined in CADLIVE

	ReactionType	Symbol	Regulator Reaction Equation
1	binding		A + B <-> A:B
2	binding with stoichiometric changes		A + B > A-B
3	homo association or modification		2A <-> A:A
4	homo association or modification with stoichiometric changes		2A > A-A or A + X > A-X
5	reversible conversion		A <> B
6	irreversible conversion		A > B
7	reversible conversion regarding multicomponents		A + B <> C + D
8	irreversible conversion regarding multicomponents		A + B > C + D
9	transport		A(cytoplasm) -> A(nucleoplasm)
10	transcription		gene(A) -> mRNA(A)
11	translation		mRNA(A) -> A
12	protein synthesis		gene(A) -> A
13	degradation		A ->
14	activation of species		A ->> B
15	inhibition of species		I -II B
16	activation of regulator		A ->> reaction process
17	inhibition of regulator		I -II reaction process

2. Separation of real reactions and nodes from a domain-level interaction map



Supplementary Figure 1 Separation of real reactions and nodes from a domain-level interaction map
(A) A domain- or subunit- based map with virtual reactions and nodes. (B) A real map containing no virtual reaction and node. CADLIVE can switch off the display of subunits, domains, virtual reactions and nodes, and InnerLink.

3. An algorithm of pathway search

We employ a multiple search tree algorithm to explore all possible pathways between a starting molecule and an end molecule, as shown in **Supplementary Figure 2**. The search employs the real reactions and nodes described by RREs. A starting molecule and an end molecule are named as *Start* and *End*, respectively. The number of all possible pathways between the *Start* and *End* species may rapidly increase with an increase in a network size, with an increase in the pathway length between the *Start* and *End* species, or with an increase in the nodes with high connectivity. To avoid such calculation explosion, we define the *MaximumLength* that determines the length of reaction pathways to search. In another option, the application allows for removing specific molecules with high connectivity such as ATP and NADH from the network.

Start and *End* are selected out of the species list in the GUI application. The multiple search tree algorithm is provided by:

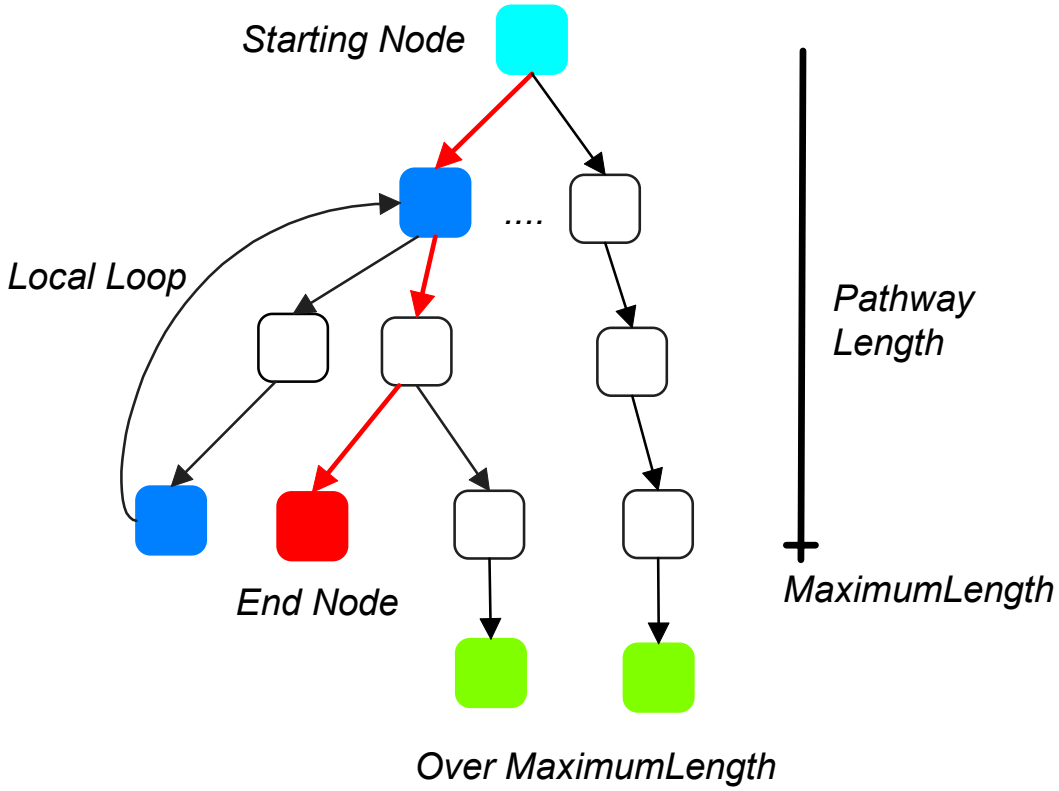
$$pathway\_search(key, End),$$

as shown in **Supplementary Figure 3**. At the beginning the key is set to *Start*. This function is recursively performed by updating the second messengers as new key species until the *END* molecule is found, a local loop is detected, or the length of the pathway sequence is over a given *MaximumLength*. This algorithm is an exhaustive search and excludes duplicated pathways.

The regulation status is defined as a relationship between the connecting species in each pathway. The regulation status is “activation” or “suppression”. If an increase in the concentration of the former species causes that of the latter species to increase, the regulation status between them is activation. When an increase in the concentration of the former species results in a decrease in the concentration of the latter species, the regulation status is suppression. Activation is annotated with a +1 sign, and suppression with a –1 sign. Multiplying all the status from a starting node to an end node provides the status of the entire pathway. The regulation status is described in detail in **Supplementary Table 2**.

We exemplify the search procedure for RREs. When a key is found in the regulator part (*R*) in an irreversible reaction ($R \; ->> \; S_1+,..+S_p \rightarrow P_1+,..+P_q$), the species in the substrate and product parts are selected as its second messengers ($S_1,..S_p,P_1,..P_q$). If the regulator is the activator, the concentration of the substrate species decreases, while that of the product species increases. The regulation status for the substrate species is suppression (-1) and that of the product species activation (+1). If the key is found in the substrate part, the species in the substrate part with the exception of itself and those in the product part are its second messengers. The regulation status for the substrate part is suppression (-1) and that for the product part is activation (+1).

If an end species is set to the same as a starting one, closed loop pathways are detected with their regulation status. If the regulation status is -1, the loop can be the negative feedback loop that provides a robust property against disturbance to the starting molecule. Conversely, if the regulation status is +1, the closed loop can be regarded as a positive feedback loop. If multiple pathways are found between a staring and end species, the robustness deriving from redundant pathways can be discussed.



Supplementary Figure 2 A multiple search tree algorithm for pathway search
The red node indicates that a target pathway is found and the red arrows form the stem pathway. The blue nodes are the same molecules, indicating that they form a local loop. The green nodes mean that the length of pathways is over a given *MaximumLength*.

void pathway _ search(*key*, *End*)
(1) int *N=0*; //The counter for the length of pathways sequence
(2) The RREs that connect the key species to its second messengers are explored;
(3) **if** (The second messengers are found from such RREs) **then**
(4) **for** (Each second messenger)
(5) **if** (The second messenger is the *END* species) **then** Store the pathways; **return**;
(6) **elseif** (A local loop is detected) **goto** stop;
(7) **elseif** (*N* > *MaximumLength*) **goto** stop; // The explored pathway is over *MaximumLength*.
(8) **else** *N*++; *pathway _ search*(*second_messenger*, *End*) ; //A recursive function is formed.
(9) **endif**
(10) stop:
(11) **endfor**
(12) **else return**; //No pathways are found.
(13) **endif**

Supplementary Figure 3 A brief flowchart of the multiple search tree algorithm for pathway search

4. An algorithm for virtual knockout

A virtual knockout algorithm is developed to delete target genes from a biochemical network and is implemented into the CADLIVE GUI network editor. A recursive algorithm is employed to trace all possible modified or catalyzed molecules from the target genes to be deleted. The nodes of a biochemical map are provided by:

$node(i),\ \ i = 1,2,...,M$,

where M is the total number of nodes. The interactions among the nodes are described by the RREs. The attributes of each node that are necessary for virtual knockout are given by:

$node(i)$ {
    N_{in} (integer);
    IsExternal (true or false);
}

where N_{in} is the number of the ingoing edges to $node(i)$ and *IsExternal* determines whether $node(i)$ with $N_{in} = 0$ is removed or not. If $N_{in} = 0$, the node is deleted as it is not produced. On the other hand, if one does not delete the node with $N_{in} = 0$, set *IsExternal* to *true*, which prevents the node with $N_{in} = 0$ from being removed. For example, *IsExternal* for the gene nodes should be set to *true*, because they should not be removed for $N_{in} = 0$. The default value of *IsExternal* is false. After setting these attributes, a target gene of $node(i)$ is removed by the recursive function:

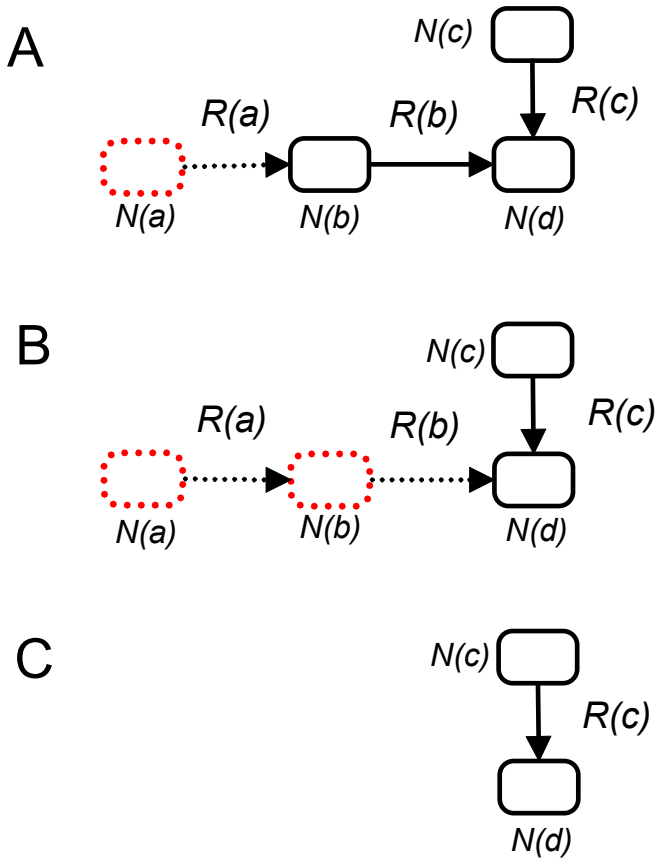
$deletion\_node(node(i))$,

as shown in **Supplementary Figure 4**. **Supplementary Figure 5** explains the process of gene knockout and the role of the presented attributes of N_{in} and *IsExternal*. The nodes and reactions are named *N* and *R*, respectively. The *IsExternal* of *N(a,b,c,d)* is set to false, false, true, and false, respectively. In the panel of A, *N(a)* is deleted, subsequently *R(a)* is deleted. Since *N(c)*.N_{in} = 0 and *N(c).IsExternal* = true, *N(c)* is protected or not deleted. In the panel of B, *N(b)* is deleted as *N(b)*.N_{in} = 0 and *N(b).IsExternal* = false, resulting in the deletion of *R(b)*. In the panel C, as *N(d)* N_{in} =1, *N(d)* is left with *N(c)*.

void deletion _ node(*Dnode*)

(1) The RREs that contain *Dnode* are explored for all RREs;
(2) **if** (Such RREs are not found) **then return**;
(3) **else**
(4) Such RREs are deleted;
(5) **for** (j=1; j ≤ M; j++)
(6) $node(j).N_{in}$ is calculated;
(7) **if** ($node(j).IsExternal = true$ & $node(j).N_{in} = 0$) **then**
(8) **return** $deletion\_node(node(j))$; // A recursive function is formed.
(9) **endif**
(10) **endfor**
(11) **endif**

Supplementary Figure 4 A brief flowchart of the algorithm for gene knockout.
Dnode is the species to be deleted. *M* is the total number of species.



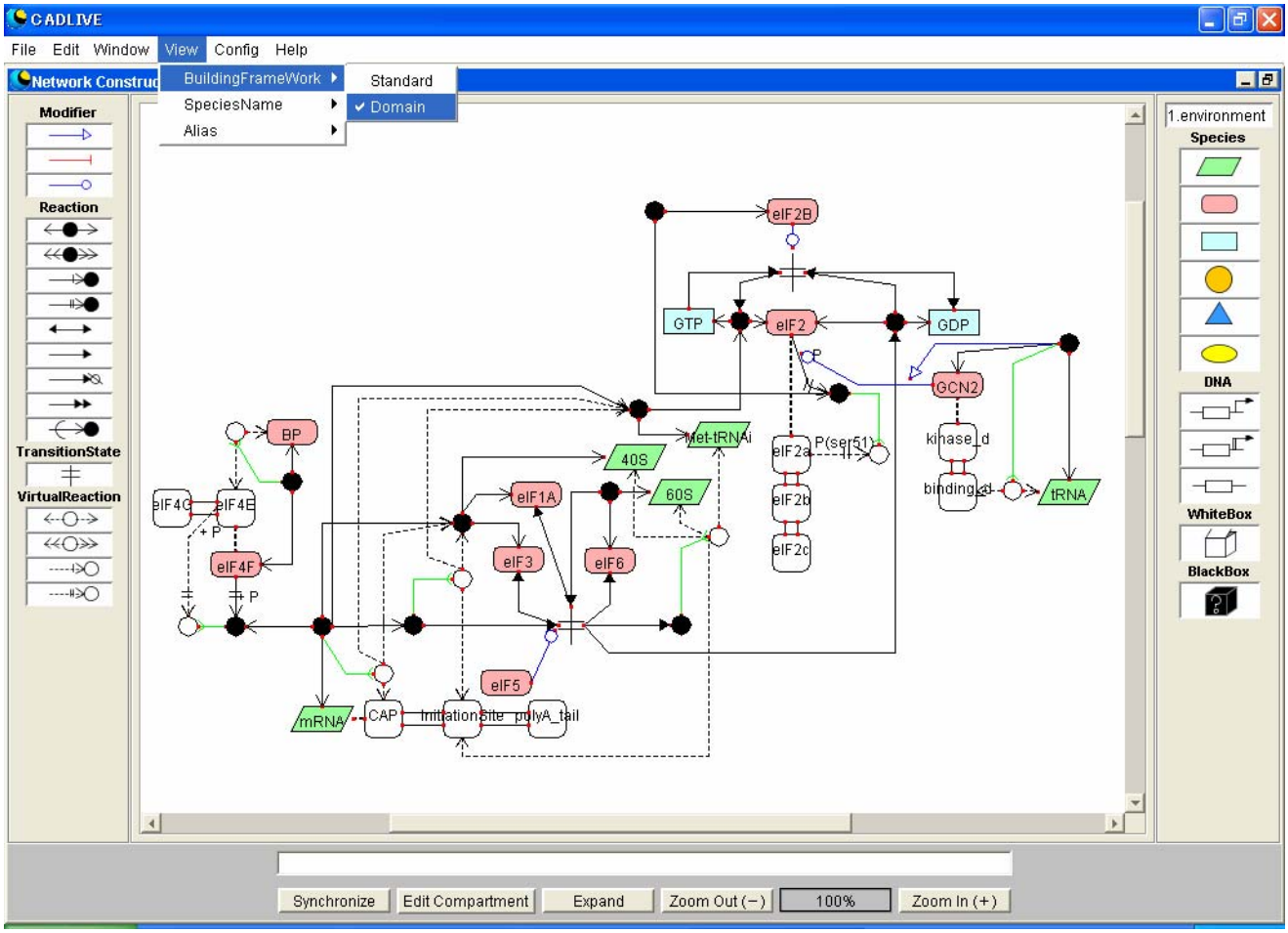
Supplementary Figure 5 A process flow of gene knockout and the role of the presented attributes
The nodes and reactions are named *N* and *R*, respectively. The *IsExternal* of *N(a,b,c,d)* is set to false, false, true, and false, respectively. (**A**) *N(a)* is knocked out, subsequently *R(a)* is deleted. Since $N(c).N_{in}$ = 0 and *N(c).IsExternal* = true, *N(c)* is protected or not deleted. (**B**) *N(b)* is deleted as $N(b).N_{in}$ = 0 and *N(b).IsExternal* = false, resulting in the deletion of *R(b)*. (**C**) As *N(d)* N_{in} =1, *N(d)* is left with *N(c)*.

5. GUI application of the extended CADLIVE

A biochemical network map is drawn by the CADLIVE GUI network constructor (**Supplementary Figure 6**). When proteins or RNAs are expanded into domains and the virtual reactions and nodes are used, one can name the domains and virtual nodes in the "FrameWork Editor" from the "Window" menu, as shown in **Supplementary Figure 7**. The "Frame Work Editor" consists of three tables: Domain, Species, and Virtual Node, where one is allowed to edit each table. If one likes to switch off the part of the domains and the Virtual reactions and nodes from the "NetworkConstructor" window (**Supplementary Figure 1**), one selects "standard" in the "Building Frame Work" from the "View" menu. Use of WhiteBox makes subnetwork modules. By choosing the "WhiteBox Tree Window" from the "Window" menu, the window of "List of WhiteBox" appears and one sees the species list of each WhiteBox, as shown in **Supplementary Figure 8**. WhiteBox is allowed to pack the related real or virtual nodes into a subnetwork by selecting them on the "Network Constructor" window. Since WhiteBox does not appear in the "Data Editor" window, WhiteBox must be edited on the menu of List of WhiteBox, where one edits the network modules.

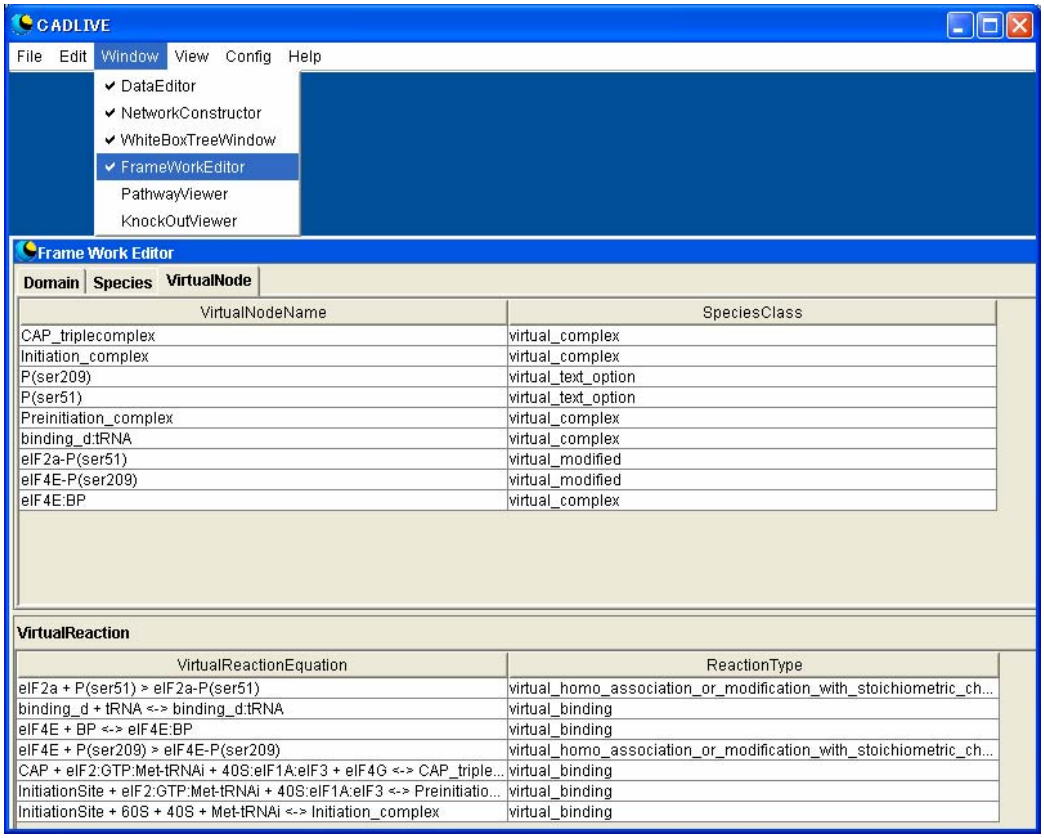
From the "Window" menu in the CADLIVE GUI editor one jumps to the pathway search module (PathwayViewer) or to the virtual knockout module (KnockOutViewer), as shown in **Supplementary Figure 9**. In the pathway viewer (**Supplementary Figure 10**), one sets the *Start* and *End* species and determines the value of *MaximumLength*. After pathway exploration, the resulting pathways appear together with the regulation status in the top left (**Supplementary Figure 10A**). One can see the details of the pathway to be selected. The table raw corresponding to the selected pathway is highlighted in blue and its details appear in the bottom left window. The red line is the main and stem pathway and the black ones are local loops, where the bottom species in black are connected to specific species within the red pathway, forming a closed loop pathway. By clicking the button of "Export" one is allowed to export the details of the search pathways into one's personal computer. By clicking the button of "BrowseNC", CADLIVE displays the selected pathway on the map, where it is marked by the thick red lines (**Supplementary Figure 10B**).

In the knockout viewer (**Supplementary Figure 11**), one sets the target genes to be deleted. The blue-highlighted species indicate the target genes. By clicking the button of "Knockout" CADLIVE carries out knockout simulation and displays the lists of the resulting deleted species and reactions (**Supplementary Figure 11A**). By clicking the "BrowseNC" button, one can see the biochemical network of the knockout mutant, where the dotted lines correspond to the deleted species and deleted reactions (**Supplementary Figure 11B**). By clicking the "BrowseNC" button, CADLIVE jumps to the pathway viewer. One sets the *Start* and *End* species to explore all possible pathways between them in the knockout mutant. This performs pathway search for the knockout mutant. By clicking the "BrowseNC" button, the resulting pathways are displayed in the red thick lines (**Supplementary Figure 11C**).

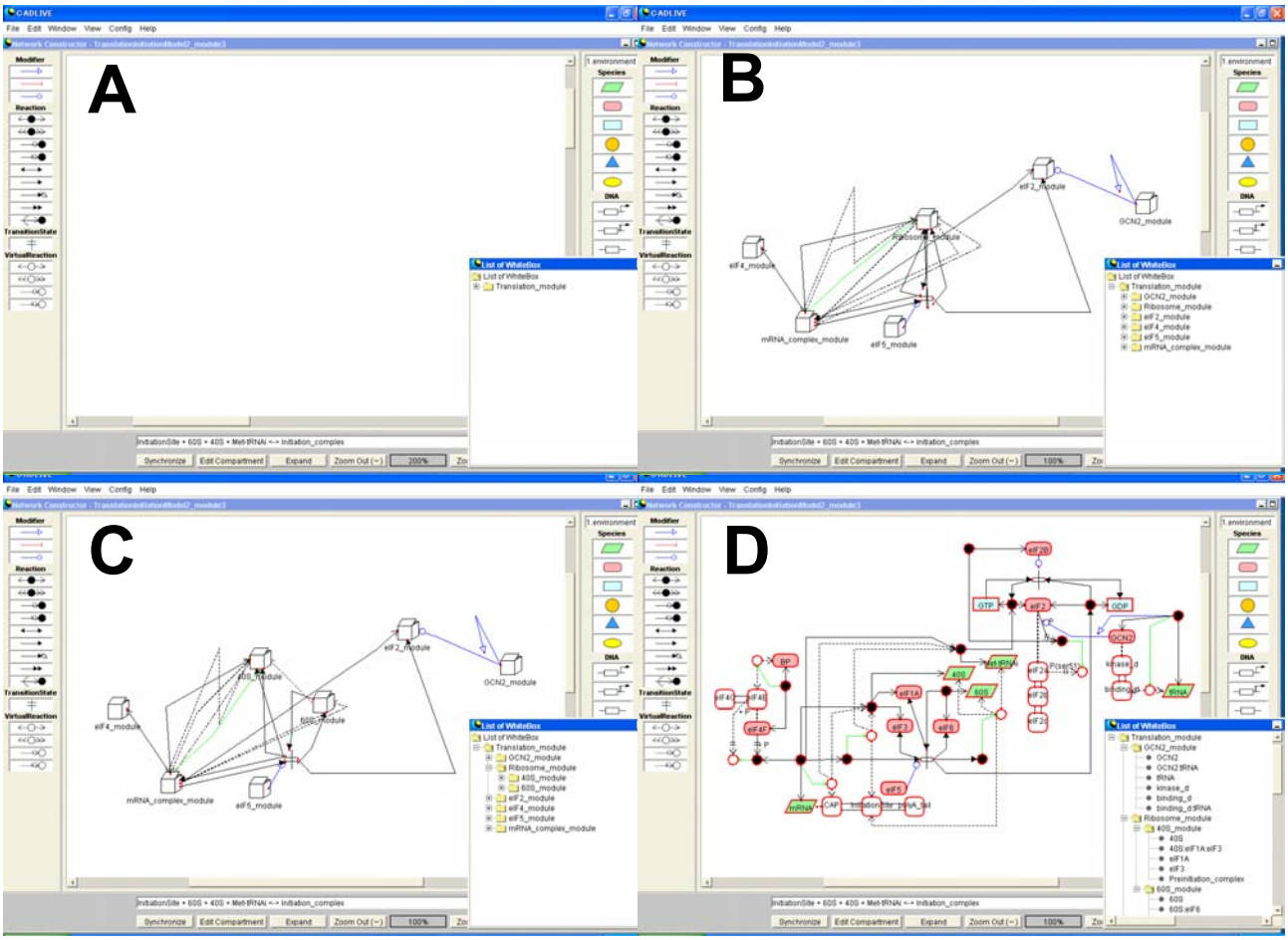


Supplementary Figure 6 A GUI network constructor of the extended CADLIVE

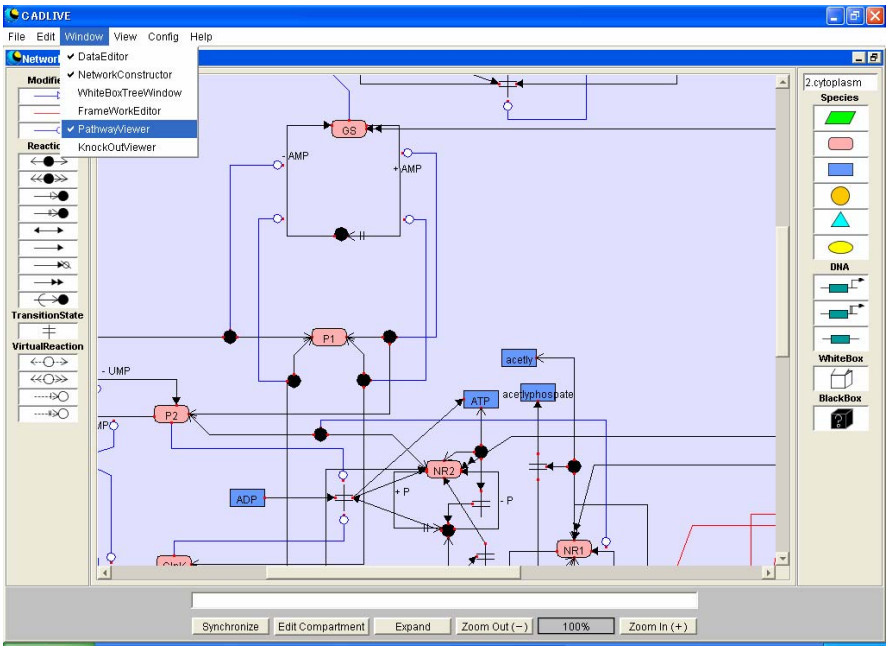
The mammalian translation initiation model is used. Details of it are described in **supplementary text 1**.



Supplementary Figure 7 "Frame Work Editor" from the "Window" menu
The "Frame Work Editor" consists of three tables: Domain, Species, and Virtual Node tables, where one is allowed to edit each table or to name the subunits or domains and virtual nodes.

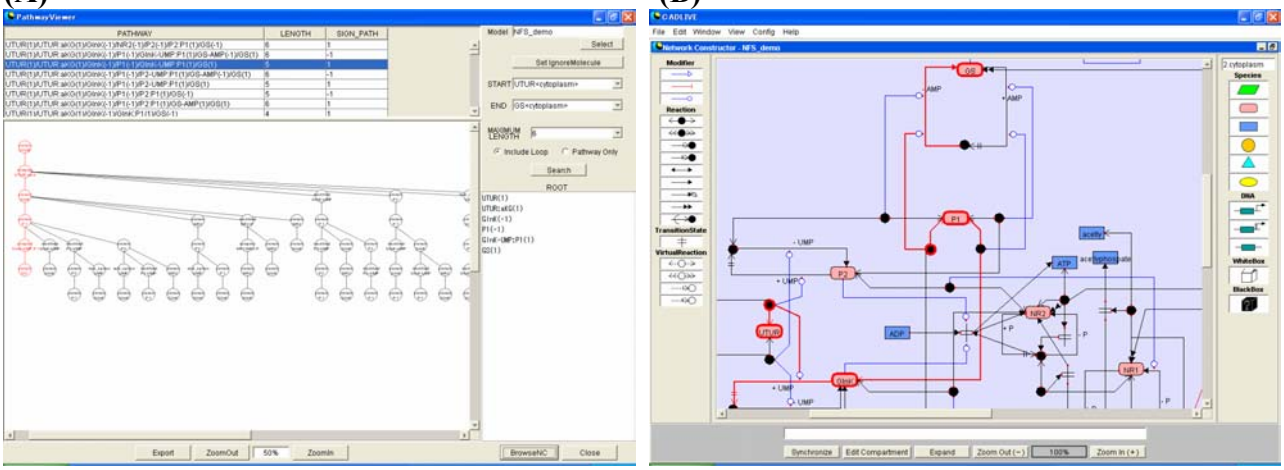


Supplementary Figure 8 A hierarchical modular architecture by using WhiteBox
One can pack molecular networks or modules into the WhiteBox. (A) The entire network. (B) Six functional modules. (C) Seven functional modules. (D) A biochemical network map at the domain-or subunit- level.



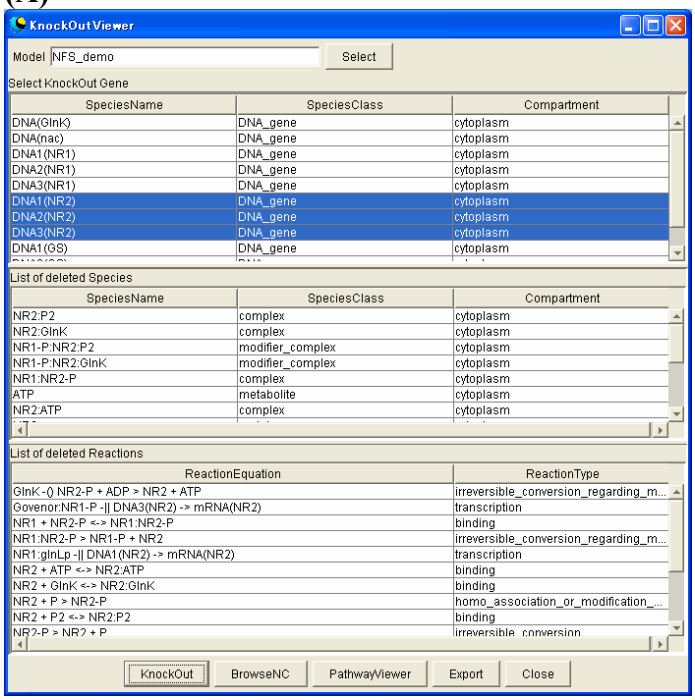
Supplementary Figure 9 Menu for movement to "Pathway Viewer" or "Knockout Viewer"
The displayed map is the *E. coli* ammonia assimilation system. The pathway search module and virtual knockout module are implemented in the extended CADLIVE.

(A) **(B)**

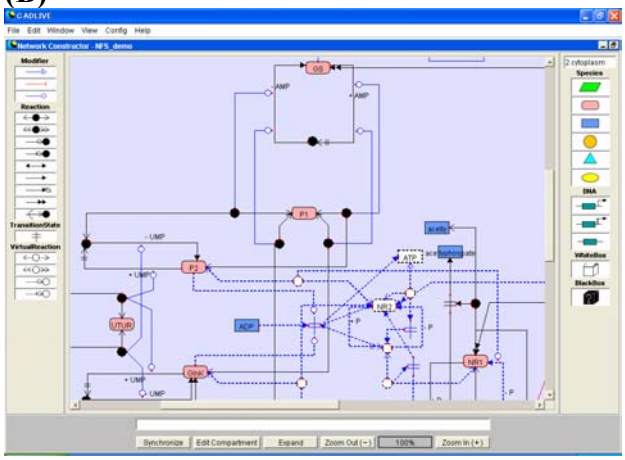


Supplementary Figure 10 Pathway viewer
(A) The window for searching the pathway between two species. The pathway of the blue-highlighted raw is displayed in the left bottom panel, where the red flow is the stem pathway and the black ones form local loops. (B) The high-lighted pathway in A is displayed in thick red lines in the CADLIVE network constructor.

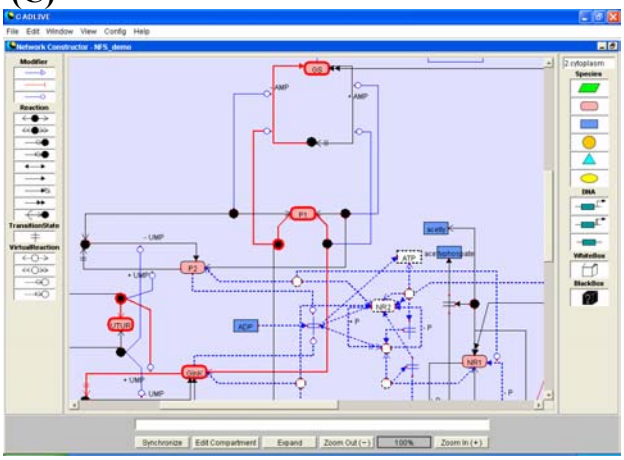
(A)



(B)



(C)



Supplementary Figure 11 Knockout viewer

(A) The window for virtual knockout. The blue-highlighted species are removed by clicking the "KnockOut" button in the bottom part. The deleted species and reactions appear in the middle and bottom tables. (B) The dotted symbols and lines are the deleted species and reactions on the network constructor. (C) Pathway viewer for the knockout mutant. When specific pathways are searched in the knockout mutant, the resulting pathways are marked in the red lines on the network constructor.

6. A translation initiation system
(1-3)

Supplementary Text 1 A summary of a mammalian translation initiation system

The 80S ribosome is dissociated into the 40S and 60S ribosomes by three eIFs: eIF1A and eIF3 bind to the 40S subunit and eIF6 to the 60S subunit, respectively. The large protein complex of eIF3 shows the ability to bind to the 40S subunit for itself, preventing the 80S subunit from associating the 40S subunit. Met-tRNAi (the initiator methyonyl-tRNA) forms a triple complex with GTP and eIF2, while the complex of eIF2:GDP does not bind to Met-tRNAi. The eIF3-bound 40S subunit promotes the triple complex of eIF2:GTP:Met-tRNAi and an mRNA to bind to the 40S subunit. An mRNA is recruited to the 40S ribosome complex in two different manners. One is the scanning mode that the 40S complex scans an mRNA from its 5' end to explore the initiation codon. The other is the internal initiation mode that the ribosome binds to an internal site of an mRNA to find the initiation codon. This report adopts the scanning mode.

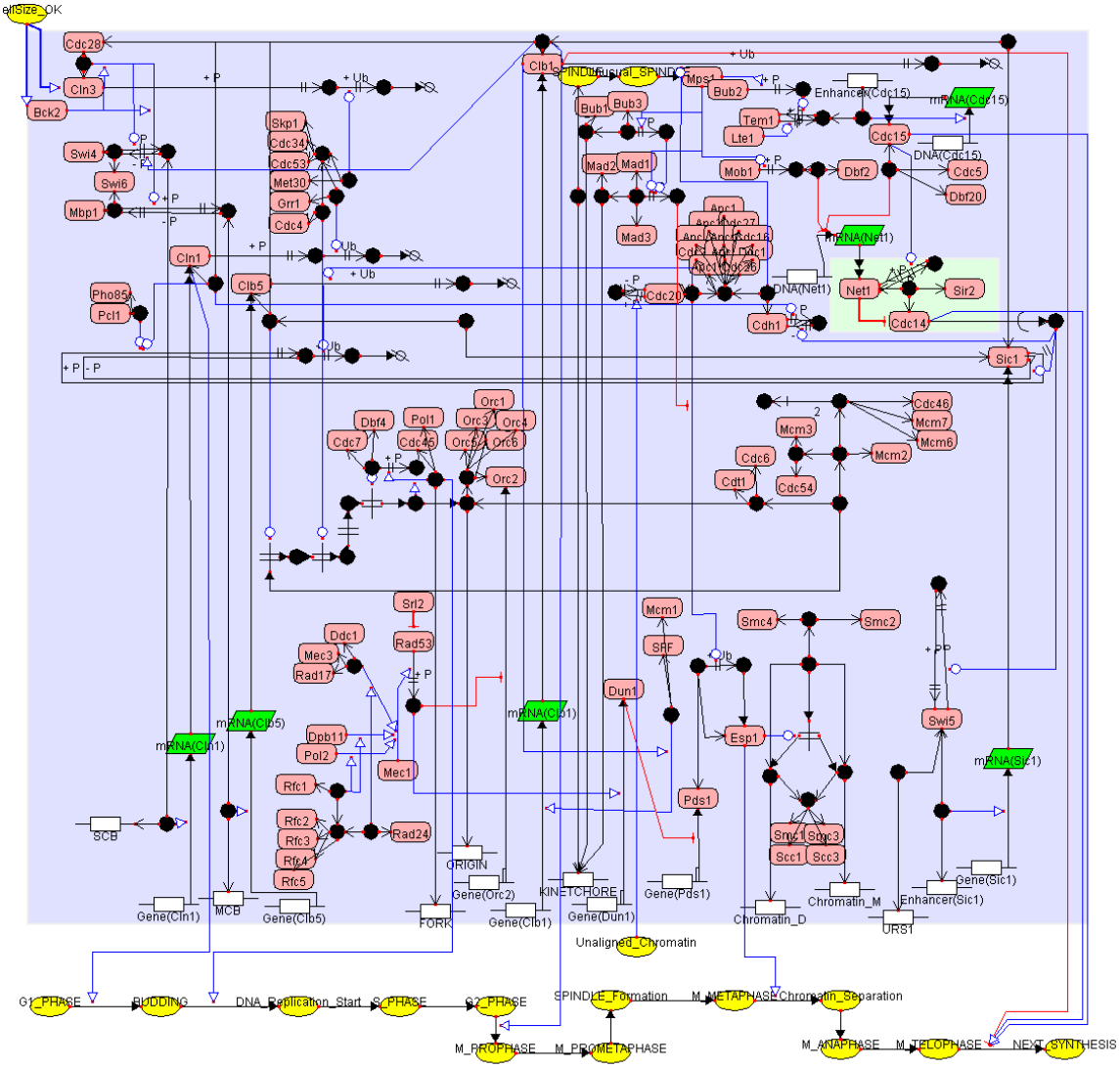
eIF4F forms a complex with eIF4A, eIF4G and eIF4E. eIF4E recognizes the cap structure of an mRNA. eIF4A has an ATP-dependent helicase activity, melting the secondary structure of the 5' end. eIF4G connects the proteins of eIF4E, eIF4A, and eIF3. The Serine 209 site of eIF4E is phosphorylated into eIF4E-P, which promotes binding to the cap to eIF4G. Consequently, the 40S ribosome complex binds to the cap structure of an mRNA through the interaction between eIF3 of the ribosome and eIF4G, forming the 40S translation initiation complex. Afterwards the 60S ribosome complex is recruited to the 40S translation initiation complex by eIF5. Simultaneously eIF2:GTP in the 40S complex is dephosphorylated and released from the 40S complex, forming the 80S translation initiation complex. To repeat translation, eIF2 must exchange GDP by GTP, which is catalyzed by eIF2B. eIF2 has three subunits and eIF2B has five subunits, respectively.

Translation initiation is suppressed by phosphorylation of eIF2. HRI, PKR (protein kinase NRA-activated), or GCN2 phosphorylates the Ser51 site of α-subunit of eIF2, forming eIF2-P. eIF2-P forms a stable complex with eIF2B, which suppresses the exchange activity of eIF2B. Since the molecular concentration of eIF2B is lower than that of eIF2, the initiation reaction stops by the entire eIF2B being bound to eIF2-P. GCN2 monitors the concentration of tRNAs by binding to them to regulate the phosphorylation activity, whereby tRNA-bound GCN2 decreases the phosphorylation activity.

7. Application of pathway search for knockout mutants to a cell cycle model

To demonstrate a feasibility of the pathway search module for knockout mutants, we simulate or predict changes in the mutant phenotypes of a budding yeast cell cycle model (4,5). To reveal a function of the target gene, molecular biology or molecular genetics experimentally investigates the phenotypes of mutants. The same investigation can be carried out *in silico* by the presented pathway search module. As shown in **Supplementary Figure 12**, CADLIVE links a series of various event species marked yellow, such as budding, DNA replication, S phase, G1 phase, M phase, spindle formation, and chromatin separation, to molecular interaction networks. Setting a target event to the end species and the G1 phase event to the start species in each single gene knockout mutant, we explored all possible pathways between the start and end species. If the pathway between them disappears, the knockout gene is shown to involve changes in the target event because the gene deletion leads to a failure of it. While varying the target event, we explored the knockout gene involving the events, thereby identifying the function of it.

We analyzed gene function by using the pathway search module for virtual knockout mutants and compared the simulated results with experimental ones. This application module simulates which event is involved in knockout genes. There are three types of simulated results: (A) Simulation determines whether the deleted genes are involved in changes in specific phenotypes; (B) Simulation does not suggest any gene function. (C) Simulated results are not consistent with experimental data. As shown in **Supplementary Table 4**, the events involved in knockout genes <*No. 1, 3, 4, 5, 6, 7, 8, 10, 11*> are almost consistent with experimental phenotypes, confirming the accuracy of the budding yeast cell cycle map. On the other hand, the pathway simulation hardly distinguishes any quantitative changes such as stop, delay and suppression. It is because the simulated result is all or nothing for each event. The behaviors for mutants <*No. 2, 12, 13*> are consistent with experimental data. In these cases, the gene function of them is not identified because the gene knockout mutants do not affect the cell cycle. The prediction of <*No. 9*> does not agree the experimental data, suggesting the biochemical network has some missing processes to be improved. While the application of the pathways search for knockout mutants is limited to simulation of which event is involved in knockout genes, the application is still useful for simulating the change in knockout mutant phenotypes or confirming the validity of network maps.



Supplementary Figure 12 A cell cycle network map of budding yeast
This map is drawn by the extended CADLIVE GUI editor. The yellow symbols are the phenomenological events. See details of the map using the CADLIVE GUI editor.

Use of CADLIVE drew the budding yeast cell cycle map that consists of 184 players and 152 reactions by using CADLIVE. The diagram can be read from left to right, *i.e.*, the cell cycle proceeds from G1 to M phase through S and G2 phases. From the bottom of the map, we draw a flow of the yeast cell cycle events, the transcription regulation, and protein signal transduction pathways. CADLIVE is able to describe not only reactions but also various events such as budding, DNA replication, S phase, G1 phase, M phase, spindle formation, and chromatin separation. This map is one of the most sophisticated images of the whole system of yeast cell cycle.

Supplementary Table 4 Simulation of changes in mutant phenotypes in the cell cycle model

(A) Simulation determines whether deleted genes are involved in changes in mutant phenotypes; (B) Simulation does not suggest any gene function; (C) Simulated results are not consistent with experimental data.

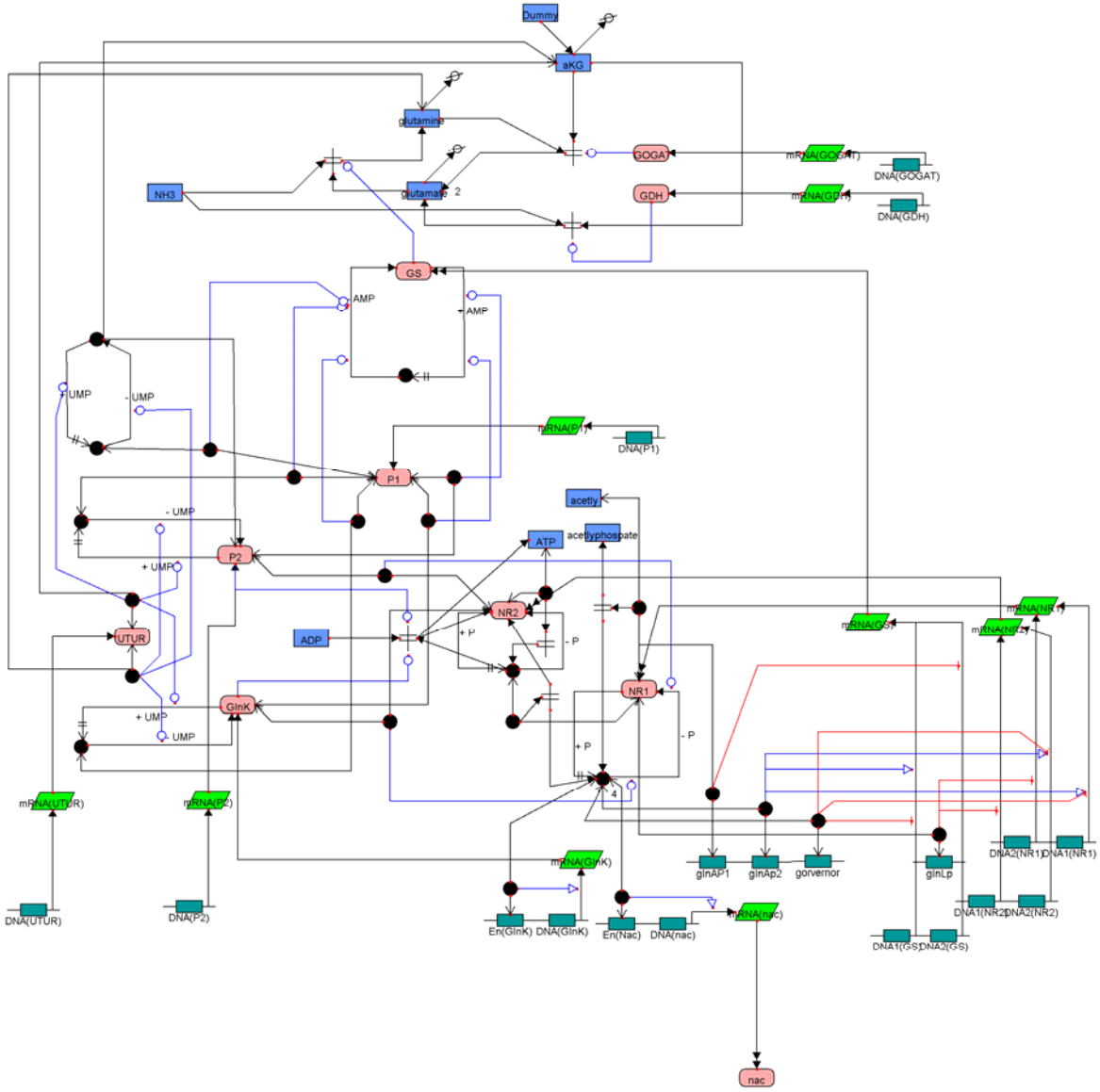
No	Knockout genes (protein name)	Simulated phenotype	Experimental phenotype	Score	Reference
1	Cln1, Cln2	The cell cycle stops before budding.	The volume is enlarged and the cycle is delayed before budding.	A	(6)
2	Clb5, Clb6	The cell cycle is completed.	S phase is delayed, while the cell cycle is completed.	B	(7)
3	Clb1, Clb2	The cell cycle stops before spindle formation.	The cell cycle stops before G2 phase. (G2 arrest)	A	(8)
4	Pds1	The cell cycle is completed.	The cell cycle is completed.	A	(9)
5	Sic1	The cell cycle does not enter G1 phase at the second cycle.	G1 phase is shortened.	A	(10)
6	Bck2	The cell cycle stops before budding.	The volume is enlarged and the cycle is delayed before budding.	A	(11)
7	Cln1, Cln2, Cln3	The cell cycle stops before budding.	The volume is enlarged and the cycle is delayed before budding. (G1 arrest)	A	(12)
8	Pds1	The cell cycle stops before chromatin separation.	Exit from M phase is delayed, while the cycle is completed.	A	(9)
9	Tem1	The cell cycle is completed.	The cell cycle stops at the end of M phase. (Telophase arrest)	C	(13)
10	Cdc14	The second cycle does not occur.	The cell cycle stops at the end of M phase. (Telophase arrest)	A	(14)
11	Mad2	The cell cycle is completed.	The cell cycle is completed.	A	(15)
12	APCs	The cell cycle is completed.	Chromatin separation is delayed, while the cell cycle is completed.	B	(16)
13	Cdh1	The cell cycle is completed.	The volume reduces and the cycle is shortened, while the cell cycle is completed.	B	(17)

8. Application of pathway search for knockout mutants to an ammonia assimilation system

To demonstrate another feasibility of the pathway search module, we predict the mechanism of how robustness is generated against gene deletions in the *E. coli* ammonia assimilation system (18-20) as shown in **Supplementary Figure 13**. The ammonia assimilation system regulates the balance between glutamine and 2-ketoglutarate. Glutamine synthetase (GS) is the key enzyme to control the glutamine concentration with respect to 2-ketoglutarate, and the activity and synthesis of GS are regulated through multiple positive and negative feedback loops.

In order to investigate the robustness of the regulation of GS activity and synthesis against gene knockout, we constructed combinatorial knockout mutants and explored the feedback pathways regulating the GS activity and synthesis, which are the glutamine-to-glutamine pathway loops. The number of all possible glutamine-to-glutamine pathways for wild type is 152 (search depth = 7), suggesting that the glutamine synthesis regulation has a highly complex and redundant structure. For single gene knockout mutants, some mutants that lack (UTUR, PI, or PII) have a small number of feedback loops; the others a large number of feedback loops, showing a robust property to deletion of any single gene, as shown in **Supplementary Table 5**.

Next we investigate the robustness of the glutamine synthesis regulation to deletion of multiple genes. Eight genes are combinatorially deleted *in silico*. For example, all possible number of mutants are calculated as ${}_8C_x$, where *x* genes are deleted. As shown in **Supplementary Figure 14**, we counted the ratio of the number of the mutants having glutamine-to-glutamine pathway loops to that of all possible combinations of multiple knockout mutants. Every single knockout mutant shows a ratio of 1, indicating that the ammonia assimilation system definitely have alternative pathways involving glutamine synthesis regulation for single gene knockout. An increase in the number of deleted genes decreases the ratio, but its decrease curve is very gentle. Some feedback loops still remain even in seven gene-deleted mutants. Compared with others systems such as the *E. coli* heat shock response and *Drosophila* circadian clock, where knockout of a few genes leads to failure of their main functions: chaperone synthesis and periodic oscillators of key proteins, the ammonia assimilation system has multiple backup pathways that buffer against the concomitant knockout of multiple genes. The glutamine synthesis has a potential network architecture that generates the robustness in face of possibly larger genetic perturbations. It can be a typical example of backup systems created by redundant pathways.



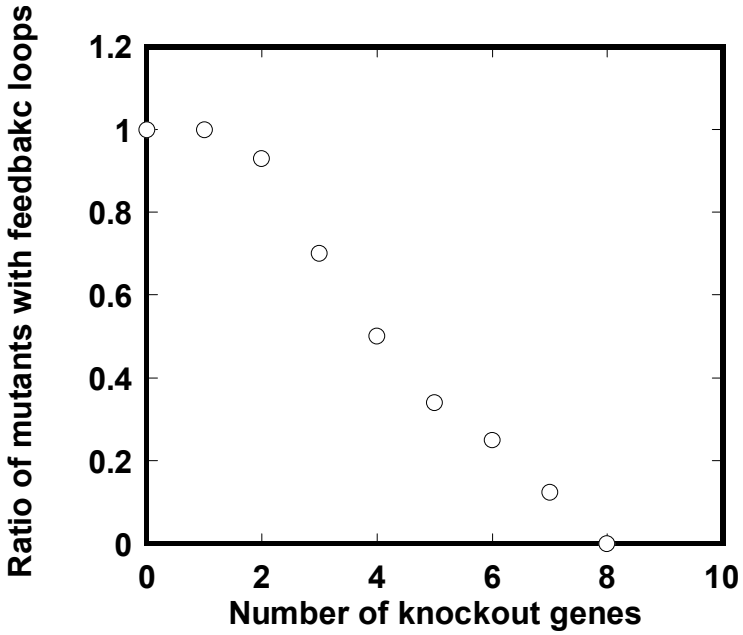
Supplementary Figure 13 A network map of the *E. coli* ammonia assimilation system
This map is designed by the extended CADLIVE network constructor.

A schematic diagram of the *E. coli* ammonia assimilation system is drawn by CADLIVE. *E. coli* absolutely needs ammonia for synthesizing glutamine and glutamate, which are the source for almost all nitrogen-containing compounds including amino acids and nucleotides. Glutamine and glutamate are synthesized through glutamine synthetase (GS), glutamate synthase (GOGAT), and glutamate dehydrogenase (GDH) by adding ammonia to 2-ketoglutarate that is an intermediate of the TCA cycle. Multiple feedback loops control the activity and synthesis of *glnA* (GS) and the transcription of nitrogen-regulated (*Ntr*) genes, *glnG* (NRI), *glnL* (NRII), and *glnK* (GlnK), whose products facilitate the adaptation to ammonia deficiency, monitoring the concentrations of glutamine and 2-ketoglutarate. The feedback loops are conventionally divided into two major loops for controlling the activity and synthesis of GS. The former module consists of UTase/UR, PII, GlnK, PI, and GS; the latter consists of UTase/UR, PII, GlnK, NRI, NRII, and GS. A central regulator of PII is the key to coordinate the two loops, which transduces the information regarding glutamine and 2-ketoglutarate to both regulations for the GS activity and synthesis.

Supplementary Table 5 Robustness in a single knockout mutant of the *E. coli* ammonia assimilation system

The number of glutamine-to-glutamine pathways is counted in each mutant that lacks a single gene. Negative and positive feedback loops indicate a regulation status of -1 and +1, respectively.

Knockout gene (protein name)	Negative feedback loops	Positive feedback loops	Total
Wild type	84	68	152
NRI	39	30	69
NRII	82	67	149
GlnK	39	30	69
UTUR	3	2	5
PI	1	1	2
PII	15	1	16
GOGAT	49	42	91
GDH	84	67	151



Supplementary Figure 14 Robustness of the glutamine synthesis regulation to combinatorial gene deletions in the *E. coli* ammonia assimilation system

The ratio of the number of the mutants having glutamine-to-glutamine feedback pathways to that of all possible combinatorial mutants is plotted with respect to the number of deleted genes. A ratio of 1 indicates that all combinatorial mutants have glutamine-to-glutamine pathways. A ratio of 0 indicates that no mutant has glutamine-to glutamine pathways.

9. Comparison of notations

Supplementary Table 6 Comparison of fundamental properties among several graphical notations

Blank indicates no implementation or no information. This comparison was carried out based on the SBGN site (http://sbgn.org/), where comparison among the notations of Kohn' MIMs, CellDesigner, PATIKA and Edinburgh are described.

Criteria	CADLIVE	Kohn's MIM	CellDesigner	PATIKA
Aim of notation	Computationally feasible and compact representation, simulation-oriented notation.	Large-scale and compact representation	Computationally feasible notations, simulation-oriented notation.	Integration, and analysis of cellular processes at molecular level. Toward public database of cellular processes.
Basic representation: temporal order of reactions or non-temporal order.	Entity-relationship diagram with the temporal order of reactions.	Entity-relationship diagram: The interactions are not limited to a reaction sequence order.	Process diagram: The temporal order of reactions is explicitly represented.	
Compact drawings: Redundant appearance of molecules (species)	Named species (molecules) and their derived species appear in only one place on a map.		Named species are allowed to repeatedly appear in complex or modified molecules, while one state only appears once on a map.	
	Redundant appearance of species is allowed when they are cofactors including ATP or NADH.			
Text representation	SBML extension		SBML	Data format is defined.
Software	CADLIVE network constructor, Pathway explorer for virtual knockout mutants, Dynamic simulator, Layout application (PC application)	No software tool	Cell Designer (Network editor) (PC application)	PATIKA Network constructor, visualizer and pathway ontology-oriented database (Client-server model)
Species entity	Many elementary species are associated with various symbols and are named.	A few elementary species are associated with a rectangle with round corners and are named.	Many types of molecular species (state nodes) are used.	Many types of molecular species (bioentity) are used.
Complex representation	A filled circle represents the complex or modified molecule.		The states of complexes are explicitly represented at the cost of an increased number of nodes and arrows.	
	Complex species are combinations or modifications of elementary species		A state node that have an N-tree data structure. Members of a molecular complex may independently participate in different	Two states are provided: 1.Each member of a complex is treated as a new state of its bioentity. 2.Members of a

			transitions.	complex may independently participate in different transitions as well as the complex itself.
Homomer	The number written on the map represents the value of N in N-mer.	Use of filled circles graphically represents the homomer.	N-mer is represented by N stacked symbols.	The number written on the map represents the value of N in N-mer.
Representation of various kinds of reactions	Many kinds of reactions (association, dissociation, reversible reactions, translocation, chemical addition, cleavage of a covalent bond, activation/Inhibition, transcription, translation, truncation, degradation, activation, and inhibition) can be drawn.			
Domain level reactions	Virtual reaction and nodes with InnerLink are presented for domain level reactions.	The arrows between domains draw domain level reactions.	Some symbols allow users to specify domains. (Small circles attached to the elementary symbols may be used.)	
Protein domains	Each protein domain is presented by a round-corner-box.	A long protein round-corner-box is employed.	Protein symbols do not allow users to specify protein domains.	
RNA domains	Each RNA domain is presented by a round-corner-box.	A long RNA round-corner-box is employed.	RNA symbols allow users to specify domains.	
Intramolecular binding	Binding between the domains within the same molecule can be described.			
Intermolecular binding	Binding between the domains in different molecules can be described.			
AND OR	It is possible to conveniently draw them by combination of the regulator arrows.	The specific symbols are available.	The specific symbols are available.	
Compartments	Compartments are represented using resizable squares or circles.	Compartments can be of any shape due to manual drawings	Compartments are represented using a round edged rectangle or an oval.	Compartments are represented by resizable simple rectangles maintaining actual neighborhood relations of real sub cellular locations.
Abstraction	Hierarchical modular structure can be constructed by WhiteBox	Manual drawings		Abstraction by nesting can be performed.

References

1. Asano, K. (2006) [Translational and transcriptional control by eIF2 phosphorylation: requirement for integrity of ribosomal preinitiation complex]. *Tanpakushitsu Kakusan Koso*, **51,** 389-398.
2. Hershey, J.W., Asano, K., Naranda, T., Vornlocher, H.P., Hanachi, P. and Merrick, W.C. (1996) Conservation and diversity in the structure of translation initiation factor EIF3 from humans and yeast. *Biochimie*, **78,** 903-907.
3. Merrick, W.C. and Hershey, J.W.B. (1996) The pathway and mechanism of eukaryotic protein synthesis. In Hershey, J. W. B., Mathews, M. B. and Sonnenberg, N. (eds.), *Translational Control*. Cold Spring Habor Laboratory Press, pp. 31-69.
4. Kurata, H., Matoba, N. and Shimizu, N. (2003) CADLIVE for constructing a large-scale biochemical network based on a simulation-directed notation and its application to yeast cell cycle. *Nucleic Acids Res*, **31,** 4071-4084.
5. Chen, K.C., Csikasz-Nagy, A., Gyorffy, B., Val, J., Novak, B. and Tyson, J.J. (2000) Kinetic analysis of a molecular model of the budding yeast cell cycle. *Mol Biol Cell*, **11,** 369-391.
6. Dirick, L., Bohm, T. and Nasmyth, K. (1995) Roles and regulation of Cln-Cdc28 kinases at the start of the cell cycle of *Saccharomyces cerevisiae*. *EMBO J*, **14,** 4803-4813.
7. Schwob, E. and Nasmyth, K. (1993) CLB5 and CLB6, a new pair of B cyclins involved in DNA replication in *Saccharomyces cerevisiae*. *Genes Dev*, **7,** 1160-1175.
8. Surana, U., Robitsch, H., Price, C., Schuster, T., Fitch, I., Futcher, A.B. and Nasmyth, K. (1991) The role of CDC28 and cyclins during mitosis in the budding yeast *S. cerevisia*e. *Cell*, **65,** 145-161.
9. Yamamoto, A., Guacci, V. and Koshland, D. (1996) Pds1p, an inhibitor of anaphase in budding yeast, plays a critical role in the APC and checkpoint pathway(s). *J Cell Biol*, **133,** 99-110.
10. Schneider, B.L., Yang, Q.H. and Futcher, A.B. (1996) Linkage of replication to start by the Cdk inhibitor Sic1. *Science*, **272,** 560-562.
11. Di Como, C.J., Chang, H. and Arndt, K.T. (1995) Activation of CLN1 and CLN2 G1 cyclin gene expression by BCK2. *Mol Cell Biol*, **15,** 1835-1846.
12. Richardson, H.E., Wittenberg, C., Cross, F. and Reed, S.I. (1989) An essential G1 function for cyclin-like proteins in yeast. *Cell*, **59,** 1127-1133.
13. Shirayama, M., Matsui, Y. and Toh, E.A. (1994) The yeast TEM1 gene, which encodes a GTP-binding protein, is involved in termination of M phase. *Mol Cell Biol*, **14,** 7476-7482.
14. Fitzpatrick, P.J., Toyn, J.H., Millar, J.B. and Johnston, L.H. (1998) DNA replication is completed in Saccharomyces cerevisiae cells that lack functional Cdc14, a dual-specificity protein phosphatase. *Mol Gen Genet*, **258,** 437-441.
15. Alexandru, G., Zachariae, W., Schleiffer, A. and Nasmyth, K. (1999) Sister chromatid separation and chromosome re-duplication are regulated by different mechanisms in response to spindle damage. *Embo J*, **18,** 2707-2721.
16. Rudner, A.D. and Murray, A.W. (2000) Phosphorylation by Cdc28 activates the Cdc20-dependent activity of the anaphase-promoting complex. *J Cell Biol*, **149,** 1377-1390.
17. Schwab, M., Lutum, A.S. and Seufert, W. (1997) Yeast Hct1 is a regulator of Clb2 cyclin proteolysis. *Cell*, **90,** 683-693.
18. Ninfa, A.J. and Atkinson, M.R. (2000) PII signal transduction proteins. *Trends Microbiol*, **8,** 172-179.
19. Ninfa, A.J., Jiang, P., Atkinson, M.R. and Peliska, J.A. (2000) Integration of antagonistic signals in the regulation of nitrogen assimilation in *Escherichia coli*. *Curr Top Cell Regul*, **36,** 31-75.
20. Kurata, H., Masaki, K., Sumida, Y. and Iwasaki, R. (2005) CADLIVE Dynamic Simulator: Direct Link of Biochemical Networks to Dynamic Models. *Genome Res*, **15,** 590-600.