Chapter 4

Solutions to Exercises

Exercise 4.1. Consider the network depicted in Figure 4.3.
a) Is the network strongly connected? Explain your answer.
b) If the network is not strongly connected, identify its strongly connected components.
c) Does the network contain loops? If so, identify them.
d) Does the network contain cycles? If so, identify all cycles.
e) Are there any feed-forward loops? If so, identify them as positive, negative, or incoherent.
f) Are there any feedback loops? Identify them. Identify their sign as positive or negative.

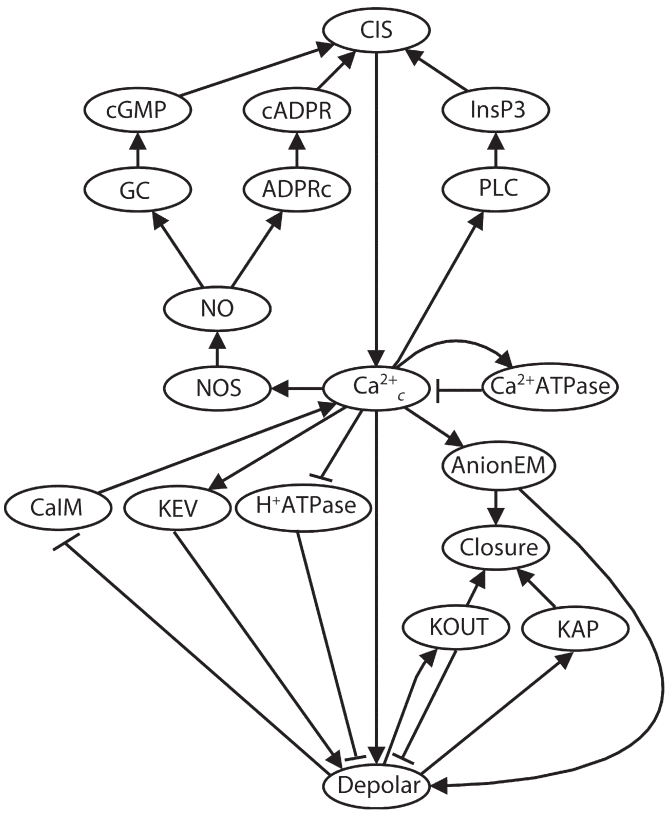


FIGURE 4.3 Figure for Exercise 4.1. The network is a part of a plant signal transduction network whose signal is the drought hormone abscisic acid and whose outcome is the closure of the stomata (microscopic pores on the leaves) [16, 17]. An arrowhead or a short perpendicular bar at the end of an edge indicates activation or inhibition, respectively. The full names of the nodes can be found in [16, 17]. Figure reprinted from Ref. [17] with permission from Elsevier.

Algebraic and Discrete Mathematical Methods for Modern Biology. http://dx.doi.org/10.1016/B978-0-12-801213-0.00023-X

Solution.

1. The network is not strongly connected. Consider, for example, the nodes H^+ATPase and KAP. A strongly connected network should have a path from H^+ATPase to KAP and from KAP back to H^+ATPase. A quick check shows that there is no path from KAP to H^+ATPase.
2. Removing the nodes Closure and KAP together with their incoming and outgoing edges gives a strongly connected network. Check that for any two nodes u and v in the remaining network, there is a path from u to v and a path from v to u.
3. The network contains no loops.
4. There are many cycles. For example, Ca^{2+}, Ca^{2+}ATPase, Ca^{2+} is a cycle of length 2; Ca^{2+}, PLC, InsP3, CIS, Ca^{2+} is a cycle of length 4. There are several more cycles that start and end at Ca^{2+} (find them!). Depolar, KOUT, Depolar is also a cycle of length 2.
5. Yes, for example, AnionEM, Depolar, KAP, Closure is a feed-forward loop because there is also a direct edge between AnionEM and Closure. This is a positive feed-forward loop. Another feed-forward loop is Ca^{2+}, H^+ATPase, Depolar (because there is also a direct edge from Ca^{2+} to Depolar). This is also a positive feed-forward loop.
6. Yes, all cycles identified in number 4 represent feedback loops. As an example, Ca^{2+}, Ca^{2+}ATPase, Ca^{2+} is a negative feedback loop, while Ca^{2+}, PLC, InsP3, CIS, Ca^{2+} is a positive feedback loop. □

Exercise 4.2. Construct the transition function and truth table for the networks in Figure 4.5. Consider both the AND and OR possibilities for the transition function of C for the network in panel (a) and for the transition function of the node B for the network in panel (b).

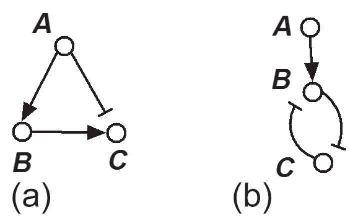


FIGURE 4.5 Figure for Exercise 4.2 and several of the follow-up exercises. Two simple signal transduction networks. For both networks, A is the source node (signal).

Solution. For the network in Figure 4.5a there are two alternative sets of transition functions:

$$\begin{aligned} f_A &= x_A, \\ f_B &= x_A, \\ f_C &= x_B \text{ AND (NOT } x_A) \end{aligned}$$

and

$$\begin{aligned} f_A &= x_A, \\ f_B &= x_A, \\ f_C' &= x_B \text{ OR (NOT } x_A). \end{aligned}$$

The truth tables for these transition functions are given in Table S4.1a.

For the network in Figure 4.5b the two alternative sets of transition functions are:

$$\begin{aligned} f_A &= x_A, \\ f_B &= x_A \text{ AND (NOT } x_C), \\ f_C &= \text{NOT } x_B \end{aligned}$$

TABLE S4.1a Truth Tables for the Transition Functions for Figure 4.5a in Exercise 4.2

x_A	f_A, f_B		
0	0		
1	1		
x_A	x_B	f_C	f'_C
0	0	0	1
0	1	1	1
1	0	0	0
1	1	0	1

and

$$f_A = x_A,$$
$$f'_B = x_A \text{ OR (NOT } x_C),$$
$$f_C = \text{NOT } x_B.$$

The truth tables for these transition functions are given in Table S4.1b. □

TABLE S4.1b Truth Tables for the Transition Functions for Figure 4.5b in Exercise 4.2

x_A	f_A	x_B	f_C
0	0	0	1
1	1	1	0
x_A	x_C	f_B	f'_B
0	0	0	1
0	1	0	0
1	0	1	1
1	1	0	1

Exercise 4.3. Show that the truth table of a Boolean function with k variables has 2^k rows and $k + 1$ columns. *Hint*: Determine the number of different sequences of length k that can be formed from 0's and 1's.

Solution. The truth table will contain a row for each combination of values for the k inputs, thus it will have every possible sequence of 0's and 1's. There are 2^k such sequences. The table will have a column for each of the k inputs, and an additional column for the output. □

Exercise 4.4. Can you guess the attractor(s) of the Boolean model in Example 4.1? Consider the cases $x_A = 0$ and $x_A = 1$ separately.

Solution. Let us first consider that $x_A = 0$. The transition functions of node B and C simplify to $f_B = f_C = 0$, thus the inevitable future state of the system is $x_A = x_B = x_C = 0$, which is therefore a steady state (also called fixed point). If we consider $x_A = 1$, the transition functions become $f_B = 1, f_C = x_B$. This indicates that the next state of node B is 1 regardless of its current state, and node C follows node B in turning on. Thus the steady state of system is $x_A = x_B = x_C = 1$. □

Exercise 4.5. Determine the state transition graph of the model in Figure 4.4 in the presence of a signal ($x_A = 1$) when using synchronous update. Compare with Figure 4.6a (right panel).

Solution. See Figure 4.6a. □

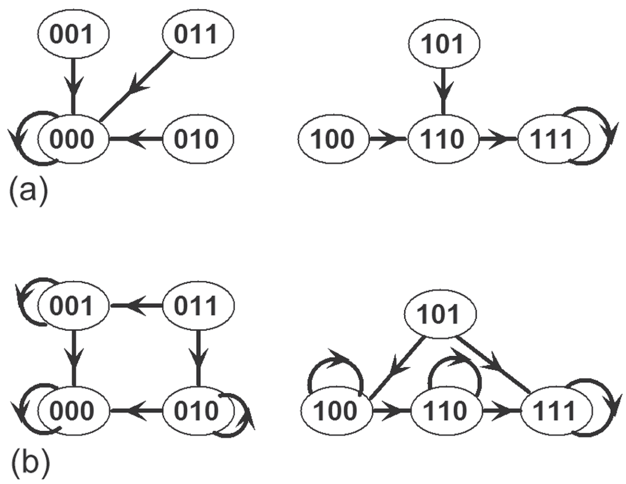


FIGURE 4.6 State transition graphs corresponding to the Boolean model presented in Figure 4.4. The symbols correspond to the states of the system, indicated in the order A, B, C; thus 000 represents $x_A = 0$, $x_B = 0$, $x_C = 0$. A directed edge between two states indicates the possibility of transition from the first state to the second by updating the nodes in the manner specified by the updating scheme. An edge that starts and ends at the same state (a loop) indicates that the state does not change during update. (a) The state transition graph corresponding to synchronous update, when all nodes are updated simultaneously. The two states that have loops are the fixed points of the system. (b) The state transition graph corresponding to updating one node at a time (general asynchronous update). While several states have loops, indicating that at least one of the nodes does not change state during update, only the two states that have no outgoing edges are fixed points of the system.

Exercise 4.6. Determine the state transition graphs for the networks in Figure 4.5, assuming synchronous update. Consider both the AND and OR possibilities for the transition function of C for the network in panel (a) and for the transition function of B for the network in panel (b).

Solution. Let us start with the network in Figure 4.5a, and the transition function for node C that contains the AND rule. We need to separately consider the cases $x_A = 0$ and $x_A = 1$. When $x_A = 0$, the transition functions for node B and C simplify to $f_B = 0, f_C = x_B$. We need to consider four initial conditions (in the order x_A, x_B, x_C): 000, 001, 010, and 011. Starting from 000, the first update leads to 000, thus this is a steady state of the system. Updating 001 yields 000. Updating 010 yields 001, and updating 011 leads to 001. The state transition graph is the first component of Figure S4.1a. When $x_A = 1$, the transition functions for node B and C simplify to $f_B = 1$, $f_C = 0$. Thus all four initial conditions (namely, 100, 101, 110, and 111) lead to the steady state 110 after a single update (see the second component of Figure S4.1a). The state transition graphs for the OR variant of the rule are also shown in Figure S4.1a and the state transition graphs for both variants of the network in Figure 4.5b are shown in Figure S4.1b. □

Exercise 4.7. Determine the state transition graph of the model in Figure 4.4 in the presence of signal ($x_A = 1$) when using general asynchronous update. Compare with Figure 4.6b (right panel).

Solution. See Figure 4.6b (right panel). □

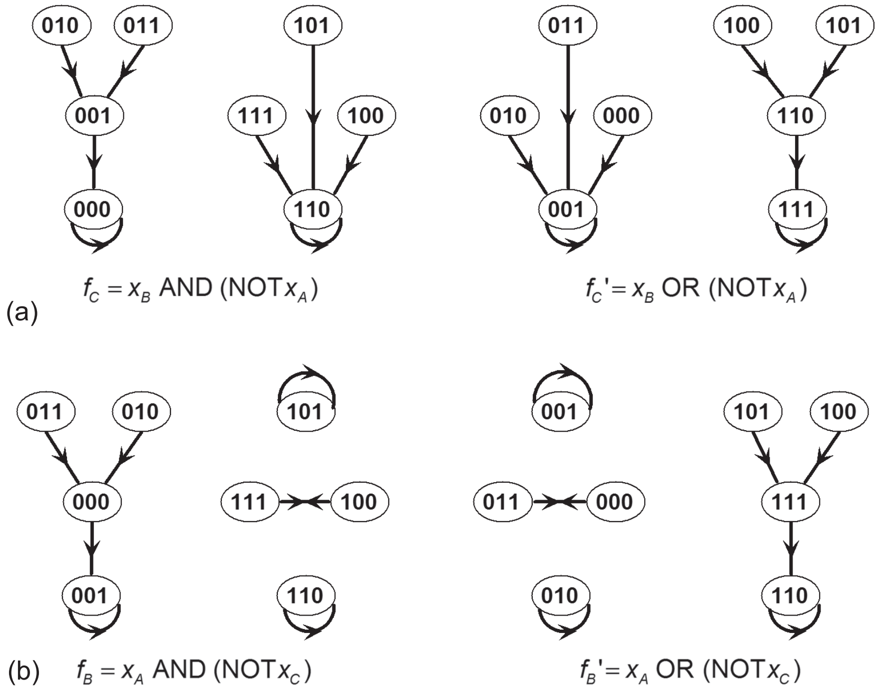


FIGURE S4.1 State transition graphs for Exercise 4.6. (a) State transition graphs for Figure 4.5a. (b) State transition graphs for Figure 4.5b.

Exercise 4.8. Determine the state transition graphs for the networks in Figure 4.5 when using general asynchronous update. Consider both the AND and OR possibilities for the transition function of *C* for the network in panel (a) and for the transition function of *B* for the network in panel (b). Consider both the sustained absence ($x_A = 0$) and presence ($x_A = 1$) of node *A*.

Solution. Because the state of the source node *A* does not change, we will not consider its update. Thus each state can have up to two successors, one for updating node *B* and one for updating node *C*. The resulting state transition graphs are shown on Figure S4.2, where (a) corresponds to Figure 4.5a and (b) to Figure 4.5b. □

Exercise 4.9. For each of the cases considered in Exercises 4.6 and 4.8, compare the steady states obtained when using synchronous update and general asynchronous update. Are the steady states the same?

Solution. Yes, the steady states are the same. □

Exercise 4.10. Consider the network in Figure 4.7. Determine the state transition graph of the network, using first synchronous update and then general asynchronous update.

Solution. Because the state of the input node *I* does not change, we can disregard its update. We need to separately consider the cases $x_I = 0$ and $x_I = 1$. For $x_I = 0$ the transition functions of nodes *A* and *B* simplify to $f_A = x_B, f_B = 0$, and the initial states to consider are 000, 001, 010, and 011. For $x_I = 1$ the transition functions simplify to $f_A = 1, f_B = 1$, and the initial states to consider are 100, 101, 110, and 111. The resulting state transition graphs corresponding to synchronous update (a) and general asynchronous update (b) are shown in Figure S4.3. □

Exercise 4.11. Determine the steady states of Example 4.1 by solving the set of equations $f_A = x_A, f_B = x_B$, $f_C = x_C$.

Solution. Plugging in the specific forms of f_A, f_B, f_C used in Example 4.1, we find the set of equations $x_A = x_A$, $x_A = x_B$, (x_A AND x_B) = x_C. Plugging the second equation into the third we obtain $x_A = x_C$. Thus the two solutions are $x_A = x_B = x_C = 0$, $x_A = x_B = x_C = 1$. Indeed these agree with the state transitions graphs in Figure 4.6. □

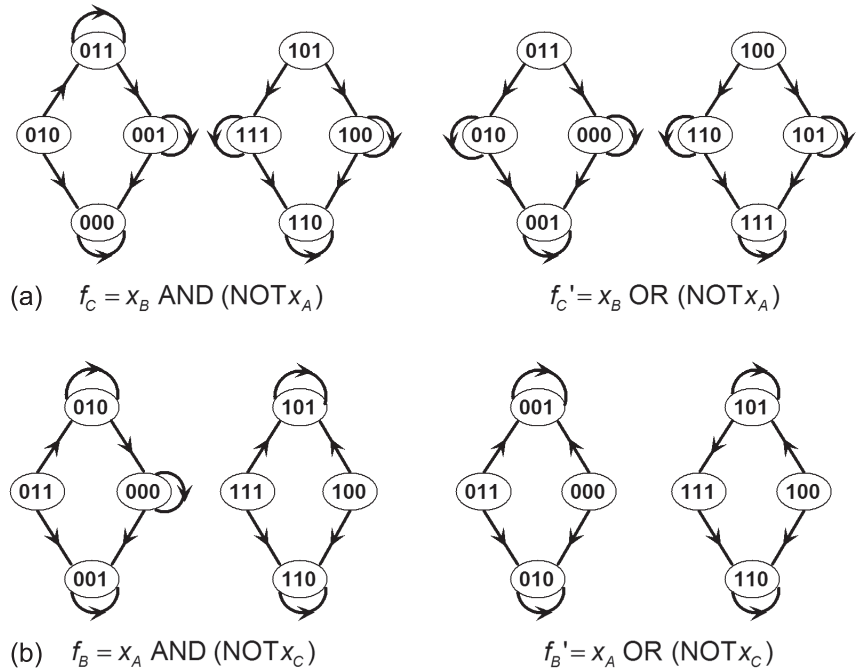


FIGURE S4.2 State transition graphs for Exercise 4.8. (a) State transition graphs for Figure 4.5a. (b) State transition graphs for Figure 4.5b.

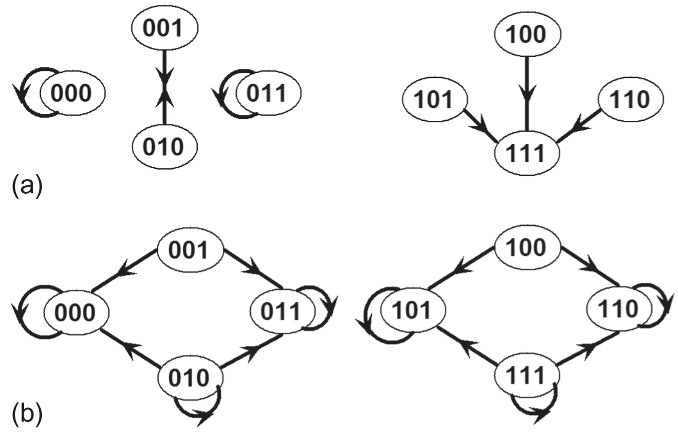


FIGURE S4.3 State transition graphs for Exercise 4.10. (a) State transition graph using synchronous update. (b) State transition graph using general asynchronous update.

Exercise 4.12. For each of the networks considered in Exercises 4.6 and 4.8,

1. Compare the complex attractors obtained when using synchronous update and general asynchronous update. Are they the same?
2. Find the basins of attraction for each of the steady states.

Solution.

1. There are no complex attractors for the network in Figure 4.5a. There is a two-state limit cycle for synchronous update of the model variant with the transition functions $f_A = 1, f_B =$ NOT $x_C, f_C =$ NOT x_B (see the second component of Figure S4.1b). This limit cycle is not preserved when using general asynchronous update, because the two states of the limit cycle differ in the state of two nodes. There is another two-state limit cycle for synchronous update of the model variant with the transition functions $f_A = 0, f_B' =$ NOT $x_C, f_C =$ NOT x_B (third component of Figure S4.1b). This limit cycle is also not preserved when using general asynchronous update.

2. The basin of attraction of a steady state includes all the states whose update eventually leads to that steady state. In the state transition graph these states are starting points of paths or loops that lead to the steady state. Inspecting the state transition graphs in Figures S4.1 and S4.2, we can summarize the basins corresponding to synchronous update and to general asynchronous update in Table S4.2. The two update methods lead to the same basins for six of the ten steady states. In the remaining cases the states that formed a limit cycle for synchronous update are in the basin of attraction of both steady states of the system for general asynchronous update.

TABLE S4.2 Summary of the Steady States' Basins of Attraction for Exercise 4.12

Case	Steady State	Synchronous Basin	Asynchronous Basin
(a) AND	000	000, 001, 010, 011	000, 001, 010, 011
(a) AND	110	110, 101, 110, 111	110, 101, 110, 111
(a) OR	001	000, 001, 010, 011	000, 001, 010, 011
(a) OR	111	100, 101, 110, 111	100, 101, 110, 111
(b) AND	001	000, 001, 010, 011	000, 001, 010, 011
(b) AND	101	101	100, 101, 111
(b) AND	110	110	100, 110, 111
(b) OR	001	001	000, 001, 011
(b) OR	010	010	000, 010, 011
(b) OR	110	100, 101, 110, 111	100, 101, 110, 111

□

Exercise 4.13. Consider the network in Figure 4.8.

1. Determine the state transition graph and the attractors of the network using synchronous update. Find the basins of attraction for each attractor.
2. Repeat part 1, now using general asynchronous update of the node states.

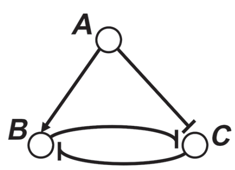


FIGURE 4.8 A simple three-node network for Exercise 4.13.

Solution. In the case of $x_A = 0$ the transition functions simplify to $f_A = 0$, $f_B = 0$, $f_C = 1$. This model is equivalent to the model depicted in Figure 4.5a when using the OR variant of the rule for node B and fixing $x_A = 0$. The state transition graph corresponding to synchronous update is shown as the third component of Figure S4.1a. The state transition graph corresponding to general asynchronous update is shown as the third component of Figure S4.2a. The basin of attraction of the steady state is indicated in the third row of Table S4.2; this basin is the same for synchronous and general asynchronous update.

In the case of $x_A = 1$ the transition functions simplify to $f_A = 1$, f_B = NOT x_C, f_C = NOT x_B. This model is equivalent to the model depicted in Figure 4.5b, when using the AND variant of the rule for node C and fixing $x_A = 1$. The state transition graph corresponding to synchronous update is shown as the second component of

Figure S4.1b. The state transition graph corresponding to general asynchronous update is shown as the second component of Figure S4.2b. The basins of attraction of the two steady states are given in the sixth and seventh rows of Table S4.2. While for synchronous update there are no states in the steady states' basin except for the steady states themselves, for general asynchronous update the two remaining states are in the basin of both steady states. □

Exercise 4.14. Determine the state transition graph for Example 4.4 when using synchronous update. Compare with Figure 4.10b. Does the steady state of *R* depend on the state of the signal *S*?

Solution. The two steady states are 000 and 110. The state of *R* is 0 in both steady states; thus, it does not depend on the state of signal *S*. □

Exercise 4.15. Consider the steady state 110 in Figure 4.10b and implement a step change in *S* from 1 to 0. Is there an excitation (state change) in *R*?

Solution. The system is in state 010 after the step change in *S*, then transitions into state 000, where it stabilizes. The state of *R* stays 0 during this trajectory, and thus there is no excitation in *R*. □

Exercise 4.16. Consider the network in Figure 4.5a.

1. Does it have a commonality with the network in Figure 4.10? Explain.
2. Consider the transition function $f'_C = x_B$ OR (NOT x_A) for node *C*. Using the synchronous state transition graphs calculated in Exercise 4.6, determine the trajectory of the system after x_A undergoes a step increase from state 001. Is there an excitation-adaptation behavior in x_C?

Solution.

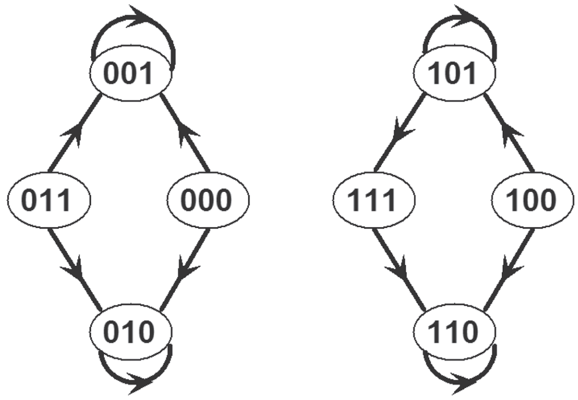
1. The networks in Figure 4.5a and Figure 4.10 both represent incoherent feed-forward loops between a source node (*A* in Figure 4.5a and *S* in Figure 4.10) and a sink node (*C* in Figure 4.5a and *R* in Figure 4.10). The difference is in the sign of the two-edge path and the direct edge between the source and sink node: in Figure 4.5a the two-edge path is positive and the edge is negative, while it is the other way around in Figure 4.10. Another difference in the transition function of the sink node is the use of the OR operator instead of AND.
$$f_A = x_A$$
$$f_B = x_A$$
$$f'_C = x_B \text{ OR } (\text{NOT } x_A)$$
2. The system is in the state 101 after the step increase for *A* is implemented. The system then transitions successively to 110 and 111, where it stabilizes. Thus the state of *C* goes from 1 to 0 and then back to 1, showing an excitation-adaptation behavior in x_C. □

Exercise 4.17. Consider the network in Figure 4.5b.

1. Does this have a commonality with the network in Figure 4.11? Explain.
2. Consider the transition function $f_B = x_A$ OR (NOT x_C) for node *B*. Using the general asynchronous state transition graphs calculated in Exercise 4.8, determine the trajectory of the system when x_A is switched to 1 from steady state 001, then switched to 0 in steady state 110. Does this system exhibit hysteresis?

Solution.

1. Yes, both are three-node networks with a single node corresponding to the signal (*S* in Figure 4.11 and *A* in Figure 4.5b) and a positive feedback loop between the remaining two nodes. The difference is that the positive feedback loop in Figure 4.5b is a mutual inhibition, while in Figure 4.11 it is a mutual activation.
2. Considering the system's steady state 001 and switching the value of the signal x_A to 1 yields state 101. The system transitions into 111 and then 110, which is a steady state (see Figure S4.4). Thus, the steady state value of *B* switches from 0 to 1. Now consider the steady state 110 and switch the value of the signal x_A to 0. This gives 010, which is also a steady state. The steady state value of *B* remains unchanged at 0 this time, showing that the system exhibits hysteresis. □



$f_B' = x_A$ OR (NOT x_C)

FIGURE S4.4 The part of Figure S4.2 relevant to Exercise 4.17.

Exercise 4.18. Consider the list of causal evidence shown in Table 4.2. Use the software NET-SYNTHESIS to construct the most parsimonious signal transduction network.

TABLE 4.2 A List of Causal Evidence Representative of What Could Be Synthesized from the Experimental Literature

Source Node	Causal Effect	Target Node or Edge	Direct Interaction?
ABA	Activates	InsPK	No
ABA	Activates	NO	No
ABA	Activates	PLD	No
NO	Activates	CIS	No
PA	Activates	ROS	No
ROS	Activates	CaIM	No
Ca^{2+}_C	Activates	AnionEM	No
Ca^{2+}_C	Activates	NO	No
PLD	Activates	PA	Yes
CIS	Activates	$Ca^{2+}{}_c$	Yes
CaIM	Activates	$Ca^{2+}{}_c$	Yes
AnionEM	Activates	Closure	Yes
KOUT	Activates	Closure	Yes
InsPK	Activates	ABA → CIS	No
Ca^{2+}_{ca}	Activates	NO → AnionEM	No
CaIM	Activates	ABA → KOUT	No

This list is derived from the work in [16] and has the same node names, but it is much simpler than the original.

Solution. Download and install the executable file drawgraph.exe from http://www2.cs.uic.edu/~dasgupta/network.synthesis/.

Then create a text file readable by NET-SYNTHESIS following the instructions shown at http://www2.cs.uic.edu/~dasgupta/network.synthesis/sample.txt.

Start drawgraph.exe and read the input file into the software. There is a short manual on the same page as the software and sample input file. The displayed network will have the nodes in random positions but nodes can be moved by holding a left-click. The first network includes only the two-node causal effects (the first 13 entries); the direct interactions are shown in blue. Select Action/Reduction to identify redundant edges and perform transitive reduction. Either version of the reduction will give the same result: One edge is removed. Select Action/Add pseudo-nodes to include and interpret the three three-node causal effects. The software indicates that nine edges and three pseudo-nodes were added. See if you can collapse some of these pseudo-nodes. Select Action/Reduction, which leads to the removal of four edges. This created pseudo-nodes with one incoming and one outgoing edge; these can be collapsed with Action/Collapse degree-2 pseudo-nodes. At this point there are no more pseudo-nodes. Check Action/Reduction again. Arrange the nodes so that you get a feel for the network. Although this exercise version is much simpler than a real abscisic acid (ABA)-induced closure network, it reflects the existence of several paths between ABA and closure and the positive feedback loop in which Ca^{2+} participates. □

Exercise 4.19. Consider the network specified by the list of edges in Table 4.3. Use the software NET-SYNTHESIS to simplify this network by designating the nodes TCR, PDGFR, NFKB and Caspase as pseudo-nodes and merging pseudo-nodes with regular nodes. An easy way to designate a node as pseudo-node in NET_SYNTHESIS is to precede its name by * (e.g., *TCR), either in the input file, or by right-clicking on the node name in the displayed network.

TABLE 4.3 The List of Edges in a Two-Signal, One Output Signal Transduction Network Used in Exercise 4.19

Source Node	Causal Effect	Target Node
Stimuli	Activates	TCR
TCR	Activates	RAS
PDGF	Activates	PDGFR
S1P	Activates	PDGFR
PDGFR	Activates	S1P
RAS	Activates	FAS
S1P	Inhibits	FAS
FAS	Inhibits	S1P
FAS	Inhibits	NFKB
FAS	Activates	Caspase
NFKB	Inhibits	Caspase
Caspase	Activates	Apoptosis

This network is derived from the T cell apoptosis signaling network displayed in Figure 4.2, and has the same node names, but it is much simpler than the original.

Solution. Create the input file in the form readable by NET-SYNTHESIS and read it. Designate TCR, PDGFR, NFKB, and Caspase as pseudo-nodes (either in the input file or on the displayed network). Use Action/Collapse degree-2 pseudo-nodes and Action/Collapse pseudo-nodes successively a few times, until no more pseudo-nodes remain. Arrange the nodes so you get a feel for the network. Although this is much simpler than the real T-cell survival signaling network, it has some of its features, such as the stimuli-induced apoptosis mediated by FAS, which is opposed by the PDGF-induced pathway. The mutual inhibition between S1P and FAS suggests the possibility of bistability, which both the biological system and the full Boolean model of the system indeed exhibit. □

Exercise 4.20. Consider the networks in Figure 4.5.

1. Assume that for the network in panel (a), an experimental observation is consistent with the steady state $x_A = x_B = x_C = 1$. What transition function does this imply for node C?
2. Assume that for the network in panel (b), an experimental observation is consistent with the steady state $x_A = x_C = 1$ and $x_B = 0$. What transition function does this imply for node B?

Solution.

1. The rules for nodes A and B are

$$f_A = x_A,$$
$$f_B = x_A,$$

with two possibilities for node C:

$$f_C = x_B \text{ AND } (\text{NOT } x_A) \text{ and } \quad f'_C = x_B \text{ OR } (\text{NOT } x_A)$$

The steady state $x_A = x_B = x_C = 1$ implies $f_A = 1$, $f_B = 1$, and $f_C = 0$, $f'_C = 1$. Thus, the experimental observation implies that the transition function for node C is $f'_C = x_B$ OR (NOT x_A).

2. The rules for nodes A and C are

$$f_A = x_A,$$
$$f_C = \text{NOT} \quad x_B,$$

with two possibilities for node B:

$$f_B = x_A \text{ AND } (\text{NOT } x_C) \text{ and } \quad f'_B = x_A \text{ OR } (\text{NOT } x_C)$$

The steady state $x_A = x_C = 1$ and $x_B = 0$ implies $f_A = 1$, $f_C = 1$, and $f_B = 0$, $f'_B = 1$. Thus, the experimental observation implies that the transition function for node B is $f_B = x_A$ AND (NOT x_C). □

Exercise 4.21. What initial conditions should be considered for Example 4.1 if we are interested in the system's response to a sustained signal?

Solution. Because node A is the signal to the network, the sustained signal corresponds to $x_A = 1$. If we have no information about the pre-stimulus state of nodes B and C, we should consider each possible state of these two nodes. Thus we should consider the initial states 100, 101, 110, and 111. □

Exercise 4.22. How many initial states should be considered for an N-node network if we have no information on the actual initial state?

Solution. A network with N nodes has 2^N different states: The number of different sequences that can be formed with entries 0 or 1 (compare with Exercise 4.3). Thus, if no information on the actual initial state is available, we should consider all 2^N different states as possible initial states. □

Exercise 4.23. Consider Example 4.1 (the network shown in Figure 4.4) when node B is knocked out (i.e., x_B is set to 0). What is the relevant state space now? Construct the state transition graph corresponding to synchronous update and general asynchronous update. Compare with Figure 4.13.

Solution. Because x_B is set to zero, we do not need to consider other states for node B. Thus the state space reduces by half. Because two of the three nodes are not updated (A is a sustained signal and B is knocked out), synchronous update is equivalent with general asynchronous update. The state transition graph is given in Figure 4.13. □

Exercise 4.24. Compare Figure 4.6 with Figure 4.13. How did the steady states of the system change due to the knockout of node *B?*

Solution. The steady states of Example 4.1 were 000 and 111. The first is preserved, as the state of node *B* was 0 in it. The second steady state is replaced by 100. Thus, although the signal is present, the activation of *C* is not possible if *B* is knocked out. □

Exercise 4.25. Determine the state transition graph of the network in Example 4.1 (Figure 4.4) for synchronous update when node *C* is knocked out.

Solution. The state space reduces to the states in which $x_C = 0$. The state transition graph is given in Figure S4.5. □

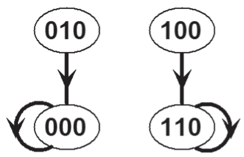


FIGURE S4.5 State transition graph corresponding to Example 4.1 (the network in Figure 4.4) when node *C* is knocked out. Because nodes *A* and *C* are not updated (*A* is a sustained signal and *C* is knocked out), synchronous and general asynchronous updates are equivalent in this case.

Exercise 4.26. Determine the state transition graph of the network in Figure 4.4 for general asynchronous update when node *C* is knocked out.

Solution. The state transition graph is the same as for synchronous update and is given in Figure S4.5. □

Exercise 4.27. Compare Figure 4.6 to your results in Exercises 4.25 and 4.26. How did the steady states of the system change due to the knockout of node *C*?

Solution. The steady states of Example 4.1 were 000 and 111. When C is knocked out, the first is preserved and the second is replaced by 110. This is closer to the original steady state than the steady state 100 obtained in the case of the knockout of *B*. □

Exercise 4.28. Consider the model of Example 4.3. As shown in Figure 4.7b, the system has two steady states, 000 and 011, when the signal is OFF ($x_I = 0$). Under general asynchronous update, both steady states are reachable from the initial conditions 001 and 010. Let's assume that steady state 000 is undesirable. Can you find a state manipulation (fixing the state of a node) such that state 000 becomes unreachable?

Solution. Fixing *B* in the ON state ($x_B = 1$) makes the transition function of *A*, $f_A = 1$, which fixes *A* in the ON state, too. This will make the steady state 000 unreachable from the initial condition 001. If *B* is fixed in the ON state, the initial condition 010 is no longer relevant. Alternatively, fixing *A* in the ON state ($x_A = 1$) fixes *B* in the ON state, making 000 unreachable from the only relevant initial condition, 010. □

Exercise 4.29. Consider the expanded network in Figure 4.15. Construct the transition functions of the nodes *A*, *B*, *C*. Construct the transition functions of the complementary nodes ∼*A*, ∼*B*, ∼*C*. Verify that the transition functions of each complementary node is the logic negation of the transition function of the respective original node.

Solution. Recall that in the expanded network all multiple inputs for a composite node are of type AND, while for the rest of the nodes multiple dependencies are of type OR. Thus, the update rules for the nodes *A*, *B*, and *C* are:

$$f_A = x_B \text{ OR } x_C,$$
$$f_B = x_A \text{ AND } (\text{NOT } x_C)\,,$$
$$f_C = x_B.$$

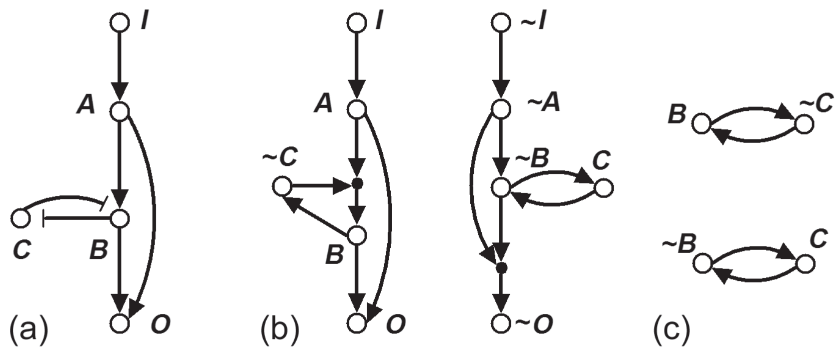


FIGURE 4.14 Illustration of methods that integrate the structure and logic of regulatory interactions. (a) A hypothetical signal transduction network. (b) The expanded representation of the network which integrates the Boolean transition functions $f_B = x_A$ AND NOT x_C, and $f_O = x_A$ OR x_B. The expanded network includes five complementary nodes, indicated by preceding the node name by $\sim$, and two composite nodes, indicated by small black filled circles. (c) The stable motifs of the expanded network in the case of a sustained input signal ($x_I = 1$). The first stable motif corresponds to the state 11101 (in the order I, A, B, C, O), while the second stable motif corresponds to the state 11011.

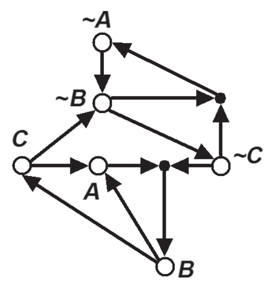


FIGURE 4.15 An expanded network for Exercise 4.29.

The transition functions for the complementary nodes are:

$$f_{\sim A} = x_{\sim B} \text{ AND } x_{\sim C},$$
$$f_{\sim B} = x_{\sim A} \text{ OR } x_C,$$
$$f_{\sim C} = x_{\sim B}.$$

To verify that the transition function of each complementary node is the logic negation of the transition function of the respective original node, we note that the state of a complementary node is always the negation of the state of the original node: NOT $x_A = x_{\sim A}$. Next, using the De Morgan law, we obtain NOT f_A = NOT (x_B OR x_C) = (NOT x_B) AND (NOT x_C) = $x_{\sim B}$ AND $x_{\sim C} = f_A$. The rest of the negations are verified in a similar way. □

Exercise 4.30. Consider the model of Figure 4.14a in the case of a sustained signal ($x_I = 1$).

1. Determine the attractors of the system under general asynchronous update. Which of these attractors corresponds to a response to the signal?
2. Set node A to OFF. Determine the attractors of the system. Did at least one attractor remain that corresponds to a response to the signal? What is your conclusion, is node A essential for the signal transduction process?
3. Set node B to OFF. Did at least one attractor remain that corresponds to a response to the signal? What is your conclusion, is node B essential to the signal transduction process?
4. Let's assume that the ON state of the output node ($x_O = 1$) is undesirable. What node interventions could make this outcome impossible?

Solution.

1. When I is ON, $f_A = f_O = 1$ and the transition rules for the rest of the network simplify to f_B = NOT x_C and f_C = NOT x_B. Thus, the only attractors of the system are the two steady states: 11101 and 11011 (with order of the variables I, A, B, C, O). Both attractors represent a response to the signal (i.e., $x_I = 1$ implies $x_O = 1$).
2. Substituting $x_A = 0$ into the transition functions leads to $f_B = 0$ and $f_O = x_B$, thus O will stabilize in the OFF state. Knocking out node A prevents the transmission of the signal. This means that node A is essential for the signal transduction process.
3. Substituting $x_B = 0$ in the transition functions leads to $f_C = 1$ and the following simplified transition function for node O: $f_O = x_A$. Having $x_I = 1$ implies $x_A = 1$, so the network stabilizes in the steady state 11011. This is one of the attractors identified in number 1, showing that knocking out node B does not prevent the transmission of the signal. Therefore node B is not essential to the signal transduction process.
4. As seen in part 2 above, if the ON state for the node O is undesirable, knocking out node A will guarantee that node O remains OFF and the signal is not transmitted. □

Exercise 4.31. Let's consider the sustained presence of the input signal ($x_I = 1$) in Figure 4.14. Simplify the transition functions of the nodes and construct the expanded network corresponding to this case. Compare with Figure 4.14c.

Solution. The sustained signal leads to $f_A = 1$, and $f_O = 1$, which leads to the stabilization of node A and O in the ON state. Nodes I, A, and O do not need to be represented in the network any longer. The regulation of B simplifies to f_B = NOT x_C and similarly f_C = NOT x_B. The expanded representation of this mutual inhibition network consists of two disconnected positive feedback loops: One formed by B and $\sim C$ and the other by $\sim B$ and C (see Figure 4.14c). □

Exercise 4.32. Determine the stable motifs of the expanded network in Figure 4.14c. Compare with the steady states you found in Exercise 4.30.

Solution. Both of the feedback loops in Figure 14c are stable motifs, the first corresponding to $x_B = 1$ and $x_C = 0$ and the second to $x_B = 0$ and $x_C = 1$. Thus there are two fixed-point attractors for this system, which differ only in the state of nodes B and C. We know from Exercise 4.29 that nodes I, A, and O stabilize in the ON state, thus the two fixed-point attractors are 11101 and 11011 (with order of the nodes I, A, B, C, O). These are the same as found in Exercise 4.28. □

Exercise 4.33. Consider again the network in Figure 4.14a, but this time consider the following update rules for the nodes B and C: $f_B = x_A$ OR (NOT x_C), and $f_O = x_A$ AND x_B.

1. Construct the expanded network.
2. Determine the elementary signaling modes between the input node I and the output node O in the expanded network
3. Determine the essential signal mediating nodes based on the elementary signaling modes.
4. Consider the sustained absence of the signal, $x_I = 0$. Determine the expanded network, its stable motifs and the corresponding steady states.

Solution. The feedback loop formed by node B and composite node $\sim C$ is a stable motif, as is the feedback loop formed by node C and composite node $\sim B$. The first motif corresponds to $x_B = 1$ and $x_C = 0$, the second to $x_B = 0$ and $x_C = 1$. Thus there are two steady states (in the node order I, A, B, C, O): 00100 and 00010.

1. The expanded network is shown in Figure S4.6a.
2. There is a single elementary signaling mode between I and O, made up by the nodes I, A, B, the composite node above O, and O. This elementary signaling mode contains both incoming edges of the composite node.
3. Node A and node B are part of the sole elementary signaling mode, so they are essential for signal mediation.
4. For $x_I = 0$ the transition functions simplify to $f_A = 0, f_B$ = NOT x_C, f_C = NOT $x_B, f_O = 0$. The states of nodes A and O stabilize, so expanded network contains the nodes B, C and the complementary nodes $\sim B$, $\sim C$. The expanded network is shown in Figure S4.6b. □

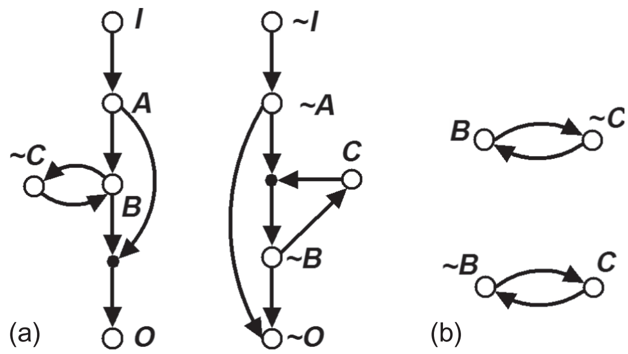


FIGURE S4.6 Expanded network for Exercise 4.33. (a) Expanded network. (b) The stable motifs of the expanded network.

Exercise 4.34. Consider the network in Figure 4.16. Construct two sets of transition functions that are consistent with this network. For each set,

1. Construct the expanded network.
2. Determine the stable motifs in the expanded network and the corresponding steady states.
3. Verify your results in point b) by determining the model's steady states analytically.

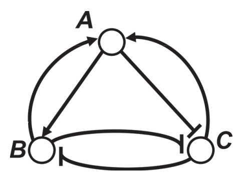


FIGURE 4.16 A simple three-node network for Exercise 4.34.

Solution. Because all three nodes have two inputs, there are eight possible sets of transition functions. We will consider two of them and will call them variant 1 and variant 2. The rest of the variants are analyzed in a similar way.

Transition functions for model variant 1:

$$f_A = x_B \text{ AND } x_C,$$

$$f_B = x_A \text{ AND (NOT } x_C),$$

$$f_C = (\text{NOT} x_A) \text{ AND (NOT } x_B).$$

Transition functions for model variant 2:

$$f_A = x_B \text{ OR } x_C,$$

$$f_B = x_A \text{ AND (NOT } x_C),$$

$$f_C = (\text{NOT } x_A) \text{ OR (NOT } x_B).$$

1. The expanded network of model variant 1 is shown in Figure S4.7a. The expanded network of model variant 2 is shown in Figure S4.7b.
2. The sole stable motif of model version 1 is the positive feedback loop between $\sim A$ and $\sim B$ (see Figure S4.7a, right panel). The corresponding steady state of nodes A and B is $x_A = x_B = 0$. Substituting into the transition function of node C yields $f_C = 1$, thus the system's steady state is 001 (in the order A,B,C).

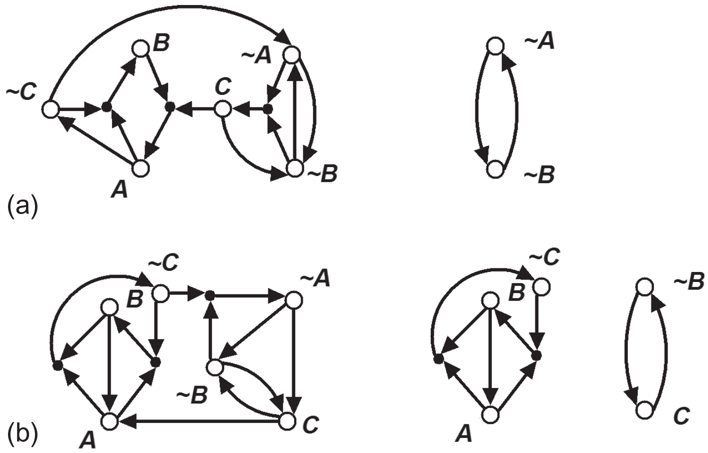


FIGURE S4.7 Expanded networks and stable motifs in Exercise 4.34. (a) The expanded network and stable motifs for model variant 1. (b) The expanded network and stable motifs for model variant 2.

Model variant 2 has two stable motifs, one formed by the nodes A, B, $\sim C$ and two composite nodes, and one formed by $\sim B$ and C (see Figure S4.7b, right panel). The first stable motif corresponds to $x_A = x_B = 1$, $x_C = 0$. The second corresponds to $x_B = 0$, $x_C = 1$, and because with those values $f_A = 1$, then $x_A = 1$. Thus model variant 2 has two steady states, 110 and 101.

3. We can determine the models' steady states by solving the set of equations $f_A = x_A$, $f_B = x_B$, $f_C = x_C$ (see Exercise 4.11).

For model variant 1, this means solving

$$x_A = x_B \text{ AND } x_C,$$

$$x_B = x_A \text{ AND } (\text{NOT } x_C),$$

$$x_C = (\text{NOT } x_A) \text{ AND } (\text{NOT } x_B).$$

Substituting the first equation into the second equation leads to $x_B = x_B$ AND x_C AND (NOT x_C) = 0. Substituting x_B into the first equation yields $x_A = 0$. The third equation then leads to $x_C = 1$. This agrees with the steady state in number 2.

For model variant 2 we need to solve

$$x_A = x_B \text{ OR } x_C,$$

$$x_B = x_A \text{ AND } (\text{NOT } x_C),$$

$$x_C = (\text{NOT } x_A) \text{ OR } (\text{NOT } x_B).$$

Substituting the first equation into the third yields x_C = (NOT (x_B OR x_C)) OR (NOT x_B) = ((NOT x_B) AND (NOT x_C)) OR (NOT x_B) = NOT x_B. Plugging this into the first equation, we obtain x_A = (NOT x_C) OR $x_C = 1$. Then the second equation yields $x_B = x_B$, which has two solutions, $x_B = 0$ and $x_B = 1$. Thus, there are two steady states, 101 and 110. These agree with the results in number 2.

We note that the state transition graph of this model variant was the subject of Exercise 4.13. □