www.pnas.org

Supplementary Information for

The effective graph reveals redundancy, canalization, and control pathways in biochemical regulation and signaling

Alexander J. Gates, Rion Brattig Correia, Xuan Wang, and Luis M. Rocha

Alexander J. Gates and Luis M. Rocha.
E-mails: a.agates@northeastern.edu and rocha@indiana.edu

This PDF file includes:

Supplementary text
Figs. S1 to S24
Tables S1 to S7
References for SI reference citations

Supporting Information Text

1. Schema Redescription Theory

A. Boolean minimization and prime implicants. The minimization of Boolean functions is a well known problem in electrical engineering and computer science. In both fields, the goal is typically to reduce the components and complexity needed to implement Boolean logic in electronic circuitry by removing redundancy from the computation of multivariate logical functions. The Quine-McCluskey algorithm (Q-M) is one of the best-known methods to remove the redundancy from logical functions (1, 2). Boolean minimization in the Q-M algorithm is achieved by uncovering the *prime implicants* (PI) of a Boolean function, i.e. an implication that resolves the logical value of the function (output) utilizing the least number of input variable states (or literals) possible. PI are thus minimal conditions (a conjunction of literals) to achieve automata state transition —minimal in the sense that the removal of any literal from the implication results in not knowing the output state (1). The first step of Q-M yields the set of all PI, or the *Blake Canonical Form* of a Boolean function—a disjunctive normal form of all PI. In our method, we represent PI by wildcard schemata, and the set of schemata that redescribe the automaton is a disjunction of all PI, equivalent to its Blake Canonical Form(3).

In the standard application of Q-M for Boolean minimization, the set of all PI is further reduced to uncover the *essential prime implicants*, which are PI that cover at least an entry of the LUT that cannot be covered by a combination of other PI. These essential PI are always needed to minimize a Boolean function. In contrast, redundant PI cover LUT entries that are already covered by a combination of essential PI. Finally, selective PI are neither essential nor redundant, and are selected at the end of Q-M to ensure coverage of all entries of the LUT. In the typical application of Boolean minimization, all redundant PI are removed and the final form of the Boolean function is a disjunction of all essential PI, plus selected PI needed to cover the entire LUT—selective PI are chosen to minimize the number of literals (maximizing wildcards).

In contrast to the engineering view of Boolean minimization, we are interested in characterizing the redundancy present in all possible interventions, even if some of those interventions can be built by combinations of other interventions. Therefore, we keep the entire set of PI, or the full Blake canonical form in schemata form. This allows us to preserve all possible *mechanisms* that change dynamical state in automata and are thus, in principle, amenable to control or perturbation interventions. In this sense, our goal is distinct from Boolean minimization of circuit design where the focus is on guaranteeing correct function computation with the least amount of wiring: to remove all redundancy in the computation of the full function. See (3) for additional details, including why the set of all PI is also necessary to infer symmetry constraints in schemata.

To appreciate why it is important to use all PI in the computation our proposed measures, consider the following example LUT of a Boolean function of $k = 3$ inputs $f(x_1, x_2, x_3)$:

$$\begin{aligned} 000 &\to 0 \\ 001 &\to 0 \\ \mathbf{010} &\to \mathbf{1} \\ \mathbf{011} &\to \mathbf{0} \\ \mathbf{100} &\to \mathbf{0} \\ \mathbf{101} &\to \mathbf{1} \\ 110 &\to 1 \\ 111 &\to 1 \end{aligned}$$

Its schema redescription, or Blake canonical form with all PI is:

$$\begin{aligned} 00\# &\to 0 \\ \mathbf{0\#1} &\to \mathbf{0} \\ \mathbf{\#00} &\to \mathbf{0} \\ \mathbf{\#10} &\to \mathbf{1} \\ \mathbf{1\#1} &\to \mathbf{1} \\ 11\# &\to 1 \end{aligned}$$

where the essential PI are shown in bold, with corresponding unique LUT entries also shown in bold in LUT. Note that a PI is essential if it redescribes at least one LUT entry that is not redescribed by any other PI (e.g. $010 \to 1$ is only redescribed by $\#10 \to 1$). From this redescription, the measures from main text eqs. 3-5 yield: $r_1 = r_2 = 3/8, r_3 = 1/4, e_1 = e_2 = 5/8, e_3 = 3/4, a_1 = a_2 = a_3 = 1/2$. If we considered the full Boolean minimization process, we would use (in this case) only the essential PI shown in bold, yielding: $r_1 = r_2 = 1/2, r_3 = 0, e_1 = e_2 = 1/2, e_3 = 1, a_1 = a_2 = 1/2, a_3 = 1$. Notably, using only the essential PI would lead to incorrect inferences about the result of possible interventions. First, the effectiveness of input x_3 would be inferred as maximal (1), when the non-essential PI $00\# \to 0$ and $11\# \to 1$ show that in reality it is possible to determine the state of f not knowing the state of x_3, or conversely, to control f with x_1 and x_2 alone (collective canalization). This can also be seen in traditional logical form; the minimized expression is $f = (x_2 \wedge \neg x_3) \vee (x_1 \wedge x_3)$, whereas adding the non-essential PI yields $f = (x_2 \wedge \neg x_3) \vee (x_1 \wedge x_3) \vee (x_1 \wedge x_2)$ (showing an extra term without x_3.)

Our analysis with all PI reveals the fact that while x_3 is more effective in determining the state of f ($e_3 > e_1 = e_2$), it has some redundancy ($r_3 = 1/4$) and is therefore not fully effective ($e_3 = 3/4$). Moreover, our formula (eq. 5 in main text) for deriving activity is only accurate when using all PI (see proof below). Indeed, the correct activity for input x_3 in this automaton is $a_3 = 1/2$ (only half the time this input changes its state leads f to change its output), not $a_3 = 1$ as using only the essential PI would have us infer. This example demonstrates that if our goal is to study which minimal interventions lead to state changes (rather than minimizing Boolean functions), we must include all PI in schema redescription.

Finally, this example also serves to highlight the difference between edge effectiveness and activity. While the latter does not distinguish the role of the three inputs ($a_1 = a_2 = a_3 = 1/2$), the former characterizes input x_3 as a little more effective at changing the state of f than inputs x_1 and x_2 ($e_1 = e_2 = 5/8, e_3 = 3/4$.) Indeed, this is the only input that does not appear as wildcard in any of the essential PI (appears always as literal), thus it is a little more effective (less redundant) than the other too. Our measures distinguish this behavior because the LUT entries (000, 001, 110, and 111) redescribed by the non-essential schemata/PI with wildcards in the x_3 position (00# and 11#), can also be redescribed by essential PI which have no wildcards in the x_3 position ($\#00, 0\#1, \#10$, and $1\#1$), and all other LUT entries are redescribed by schemata that also do not have wildcards in the x_3 position. In contrast, in the case of input variables x_1 and x_2, there are two (our of eight) LUT entries for each than can only be redescribed by essential PI/schemata that have wildcards in the x_1 or x_2 positions: 010 and 100 redescribed by #10 and #00 for x_1, and 011 and 101 redescribed by 0#1 and 1#1 for x_2, respectively. In summary, there are more possibilities (PI) to intervene on the state of function f that must include x_3 than is the case for the other two inputs, which are thus a little more redundant as measured by our collective canalization measures, but not by activity. Naturally, the node-level measures of collective canalization (eqs. 1-2 in main text) also paint a more accurate description of redundancy than sensitivity: $k_r(f) = 1, k_e(f) = 2, s(f) = 3/2$. As can be seen by the PI/schemata, one always needs two inputs to determine the state of f, so only one is redundant on average and the effective connectivity of the automaton is two inputs on average. In contrast, because it does not account for collective canalization ($k_c(f) = 1/2$), sensitivity posits that f is sensitive to only 1.5 inputs.

B. Aggregation of prime implicant influence. The per-node and per-edge measures of redundancy given by eqs 1 and 3 in main paper, aggregate the redundancy of PI/schemata that redescribe each entry of a LUT for the entire function or per input, respectively. The idea is to tally the redundancy of inputs of an automaton as conveyed by every PI/schema (as a possible intervention strategy). In the analysis pursued here, the aggregation is computed via the average operator (avg). This is the same as assuming that any PI/schema is a viable intervention possibility to change the state of automaton x. In (3) it was shown that the aggregation for input redundancy is bound by using minimum and maximum operators. The lower (upper) bound, obtained by substituting min (max) for avg in eq. 1, assumes that the schema that bests redescribes a given LUT entry is the one with least (most) number of wildcards.

Because we have no reason to prioritize a PI/schema over another one that redescribes the same LUT entry, we use the average operator, thus assuming that all PI are equally likely and important—as shown above, it is important to consider the influence of all PI as possible interventions. Moreover, averaging over all possible schemata, allows the additive per-edge separation of redundancy necessary for the effective graph (eqs. 3-4), whereby $k_r(x_i) = \mathrm{SUM}_j r_{ji}$ and $k_e(x_i) = \mathrm{SUM}_j e_{ji}$, as well as the clear relationship with activity and sensitivity (see proof below). One could consider a general case where a probability or weighting of each PI is considered, e.g. assigning greater importance to essential PI, but at this point no advantage to pursuing such a route was identified. The CANA package (4) defaults to calculating input redundancy and effective connectivity using the average operator, but has the available functionality to use their lower or upper bounds as well.

C. Activity and sensitivity via prime implicants. Here we show that the activity of input x_j to automaton x_i can be defined in terms of the schema redescription (the set of all PI) of x_i by eq. 5 in main text with the max operator in place of the average operator (as used for edge effectiveness in eq. 3 in main text). To see this, let us review the definition of activity (5): $a_j(x_i) = P(\overline{x_i^{t+1}}|\overline{x_j^t})$, which is the probability that automaton x_i flips its state at $t+1$ when its input x_j flips its state at t, given a uniform distribution of input states at t. To compute the activity from the LUT of x_i we consider that for every entry f_α in the LUT F_i, if flipping the state of input x_j (flipping the jth bit) leads x_i to change its state output, then that entry will add $1/|F_i| = 1/2^k$ to the activity. Summing over all LUT entries we have (5):

$$a_j(x_i) = \frac{1}{2^k} \sum_{f_\alpha \in F_i} \frac{\partial f_\alpha}{\partial x_j} \qquad [1]$$

where $\partial f_\alpha / \partial x_j = 1$ if flipping the jth bit of f_α changes x_i, and 0 otherwise.

Theorem 1.

$$a_j(x_i) = \frac{1}{|F_i|} \sum_{f_\alpha \in F_i} \frac{\partial f_\alpha}{\partial x_j} = \frac{1}{|F_i|} \sum_{f_\alpha \in F_i} \left(1 - \max_{\upsilon : f_\alpha \in \Upsilon_\upsilon^i} \big(j \rightarrowtail \# \big)_\upsilon \right) = 1 - \frac{1}{|F_i|} \sum_{f_\alpha \in F_i} \max_{\upsilon : f_\alpha \in \Upsilon_\upsilon^i} \big(j \rightarrowtail \# \big)_\upsilon \qquad [2]$$

where $(j \rightarrowtail \#)_\upsilon$ is a logical condition that is 1 / True if input x_j is a wildcard in schema f'_υ, and 0 / False otherwise.

Proof. To prove the equality of equation 2, we have to show that:

$$\frac{\partial f_\alpha}{\partial x_j} = 1 - \max_{\upsilon : f_\alpha \in \Upsilon^i_\upsilon} \left(j \rightarrowtail \#\right)_\upsilon \qquad [3]$$

which can rephrased as an existence statement:

$$\frac{\partial f_\alpha}{\partial x_j} = 0 \iff \exists \upsilon : f_\alpha \in \Upsilon^i_\upsilon \wedge \left(j \rightarrowtail \#\right)_\upsilon, \quad f_\alpha \in F_i. \qquad [4]$$

In other words, LUT entry $f_\alpha \in F_i$ contributes 0 to a_{ji} *if and only if* it is redescribed by at least one schema f'_υ with a wildcard in its x_j position. Rephrased in terms of the PI, we have that LUT entry $f_\alpha \in F_i$ contributes 0 to a_{ji} *if and only if* it is covered by at least one PI with no literals for the x_j logical variable.

Proving this statement, requires us to prove 3 simple lemmas.

Lemma 1: If LUT entry $f_\alpha \in F_i$ is redescribed by at least one schema/PI f'_υ with a wildcard in its x_j position, it will contribute 0 to activity ($\partial f_\alpha / \partial x_j = 0$).

The proof of the lemma follows from the definition of partial derivative of an automaton (eq. 7 in main text). Assume that schema/PI f'_υ redescribes $f_\alpha \in F_i$ and has a wildcard in its x_j position. Then the state of x_j does not influence the configuration of literals (other inputs) f'_υ specifies and cannot contribute to changing the state of x_i.

Lemma 2, the converse proposition of Lemma 1: if LUT entry $f_\alpha \in F_i$ contributes 0 to the activity of input x_j to automaton x_i ($\partial f_\alpha / \partial x_j = 0$), then there must exist at least one schema/PI of x_i, $f'_\upsilon \in F'_i$, which redescribes f_α and has a wildcard in its x_j position, that is, $f_\alpha \in \Upsilon^i_\upsilon$.

For an LUT entry $f_\alpha = s_1 ... s_j ... s_k$ to have $\partial f_\alpha / \partial x_j = 0$, then we must have $(s_1 ... 0 ... s_k \rightarrow s) \wedge (s_1 ... 1 ... s_k \rightarrow s)$, which is equivalent to $(s_1 ... \# ... s_k \rightarrow s)$, where s is any truth value (or literal for the input variables). Though the implicant $(s_1 ... \# ... s_k \rightarrow s)$ might not be a PI itself, Lemma 3 below says there must be a schema/PI with a wildcard in position x_j.

Lemma 3: if LUT entry $f_\alpha \in F_i$ can be covered by an implicant with a wildcard in its x_j position, then there must be at least one schema/PI having a wildcard in its x_j position.

This is straightforward considering the definition of prime implicant. Merging this implicant with other implicants to create a prime implicant will only add new wildcards without removing any existing wildcard. If the implicant cannot be covered by any prime implicant, then it is itself a prime implicant.

Combining all three lemmas, we prove our theorem.

The reader can notice that lemma 3 is contingent on using all possible PI. If only essential PIs were used, lemma 3 will not hold true as the example LUT in A shows.

2. Systems Biology Models

A. Cell Collective Data set. All experimentally-validated biochemical regulation and signalling BN models were retrieved from the Cell Collective (6) as of Aug 5th 2020 or were retrieved from literature and implemented in CANA(4). Table S1 shows the complete list of BN, including their respective Cell Collective and PubMed identifiers.

Note that two of the BN studied here contain a Boolean automata that is a full contradiction. That is, the Boolean automata is constant and does not depend on any of its inputs, despite having 4 or 8 specified inputs in its logical transition function. The presence of such logic irregularities further emphasizes the need for the effective connectivity methods presented here.

B. Existence of Fully Redundant Interactions. We found that 17 of the Cell Collective models (22%) contained at least one fully redundant edge, with 87 fully redundant edges in total. One possible explanation for the prevalence of fully redundant edges in gene regulatory or protein-interaction networks is that these models are often inferred from experimental data via information-theoretic measures, e.g. mutual information or transfer entropy (69), that can fail to discriminate between dyadic and polyadic relationships (70), and can thus miss the true multivariate dependency structure. But perhaps the main reasons are: 1) an incomplete record of experimental observation, 2) integration of experimental studies conducted by many different teams in different scenarios, and 3) modeling decisions about conflicting or weak evidence. This is especially problematic when the number of possible input combinations (k) is large and experimentally testing all possible control conditions becomes unfeasible. Thus, fully redundant edges may be included because they refer to interactions that were not fully observed. Moreover, interactions observed in a given experimental setup may be subsequently rendered redundant when considering additional experimental controls or different thresholds for interaction significance; the reverse is also possible, whereby a redundant interaction is subsequently considered necessary with additional experimental evidence or change in criterion for strength of interaction.

This can be appreciated using the example in Figure 1 in main text. Let us imagine that it is assembled by integrating distinct interaction inference studies. Suppose that it describes how genes x_1, x_2, and x_3 regulate the expression of gene x_4. Imagine that one interaction study does not include the effect of x_3, revealing a rather strong interaction relationship: $x_4 = x_1 \wedge x_2$. A separate study, on the other hand, does not control for the effect of x_1, observing not as clear an interaction effect of x_2 and x_3 on x_4. The study may reveal unequivocally that $x_2 = 0 \Rightarrow x_4 = 0$ (per LUT entries f_1, f_2, f_5 and f_6 in Figure 1 in main text). But because x_1 is not controlled in the study, correlational inference (e.g. via information-thererical measures (69)) is more uncertain as to how the expression of x_2 and x_3 affect the expression of x_4. It could be that x_1 was

Table S1. The 78 Boolean network models used in the analysis. Network names appear *ipsis litteris* for BN obtained from the Cell Collective.

	Network	Cell col. ID	PMID	Ref
1	Thaliana flower development	-	15486106	(7, 8)
2	budding yeast cell cycle	-	15037758	(9)
3	ER+ breast cancer signal transduction	-	29623959	(10)
4	Signal Transduction in Fibroblasts	1557	18250321	(11)
5	Signaling in Macrophage Activation	1582	18433497	(12)
6	Mammalian Cell Cycle	1607	19118495	(13)
7	FA BRCA pathway	1778	22267503	(14)
8	HGF Signaling in Keratinocytes	1969	22962472	(15)
9	Cortical Area Development	2035	20862356	(16)
10	Death Receptor Signaling	2084	20221256	(17)
11	Yeast Apoptosis	2135	23233838	(18)
12	Cardiac development	2136	23056457	(19)
13	Guard Cell Abscisic Acid Signaling	2161	16968132	(20)
14	T Cell Receptor Signaling	2171	17722974	(21)
15	Cholesterol Regulatory Pathway	2172	19025648	(22)
16	T-LGL Survival Network 2008	2176	18852469	(23)
17	Neurotransmitter Signaling Pathway	2202	17010384	(24)
18	IL-1 Signaling	2214	21968890	(25)
19	Differentiation of T lymphocytes	2215	23743337	(26)
20	EGFR & ErbB Signaling	2309	19662154	(27)
21	IL-6 Signalling	2314	21968890	(25)
22	Apoptosis Network	2329	19422837	(28)
23	Body Segmentation in Drosophila 2013 (We uploaded this)	2341	23520449	(3)
24	B cell differentiation	2394	26751566	(29)
25	Mammalian Cell Cycle 2006	2396	16873462	(30)
26	Budding Yeast Cell Cycle	2404	23049686	(31)
27	T-LGL Survival Network 2011	2407	22102804	(32)
28	Budding Yeast Cell Cycle 2009	2423	19185585	(33)
29	Wg Pathway of Drosophila Signalling Pathways	2663	23868318	(34)
30	VEGF Pathway of Drosophila Signaling Pathway	2667	23868318	(34)
31	Toll Pathway of Drosophila Signaling Pathway	2668	23868318	(34)
32	Processing of Spz Network from the Drosophila Signaling Pathway	2669	23868318	(34)
33	Cell Cycle Transcription by Coupled CDK and Network Oscillators	2681	18463633	(35)
34	T-Cell Signaling 2006	2691	16464248	(36)
35	BT474 Breast Cell Line Long-term ErbB Network	2697	24970389	(37)
36	HCC1954 Breast Cell Line Long-term ErbB Network	2698	24970389	(37)
37	BT474 Breast Cell Line Short-term ErbB Network	2699	24970389	(37)
38	HCC1954 Breast Cell Line Short-term ErbB Network	2700	24970389	(37)
39	SKBR3 Breast Cell Line Short-term ErbB Network	2701	24970389	(37)
40	SKBR3 Breast Cell Line Long-term ErbB Network	2703	24970389	(37)
41	HIV-1 interactions with T Cell Signalling Pathway	2738	25431332	(38)
42	T cell differentiation	2901	16542429	(39)
43	Influenza A Virus Replication Cycle	3481	23081726	(40)
44	TOL Regulatory Network	3491	23171249	(41)

Continues on the next page

(unknowingly) more frequently expressed when testing condition $x_2 = x_3 = 1$ (matching f_8 more often than f_4), but was more frequently inhibited when testing condition $x_2 = 1 \wedge x_3 = 0$ (matching f_3 more often than f_7). This would lead the study to conclude that the most likely interaction is represented by $x_4 = x_2 \wedge x_3$, though with an observed weak effect due to how frequently x_1 was expressed or not in the study for each condition. The weak effect would likely be reported and attributed to unknown causes of the expression of gene x_4—since x_1 was not controlled in this second hypothetical study.

When systems biologists synthesize both studies into a network model, such as that of in Figure 1 in main text, decisions must be made. They may consider that the correct relationship is $x_4 = (x_1 \wedge x_2) \vee (x_2 \wedge x_3)$—This would result in the same LUT as Figure 1 in main text, except $f_4 \equiv 011 : 1$. But because the reported effect of the second experiment (second term) is much weaker than the observed effect in the first experiment (first term), the modelers may opt to characterize the synthesis of both experiments as $x_4 = (x_1 \wedge x_2) \vee (x_1 \wedge x_2 \wedge x_3)$, assuming that the observed unknown factor in the second experiment is x_1 which was tested in the first experiment—because they have no evidence of any other genes being involved in the expression of x_4. This decision would lead to the example in Figure 1 in main text, which by associativity and absorption is simply $x_4 = x_1 \wedge x_2$, rendering x_3 a redundant input.

A scenario that leads to a similar result is imagining a experimental study that does control for all three input genes, but

Table S1 - *Continued from previous page*

	Network	Cell col. ID	PMID	Ref
45	Bordetella bronchiseptica	3492	22253585	(42)
46	Trichostrongylus retortaeformis	3493	22253585	(42)
47	HH Pathway of Drosophila Signaling Pathways	3506	23868318	(34)
48	B bronchiseptica and T retortaeformis coinfection	3509	22253585	(42)
49	FGF pathway of Drosophila Signalling Pathways	3510	23868318	(34)
50	Glucose Repression Signaling 2009	3511	19144179	(43)
51	Oxidative Stress Pathway	3512	23134720	(44)
52	CD4 T cell signaling	3521	25538703	(45)
53	Colitis-associated colon cancer	4601	26446703	(46)
54	Septation Initiation Network	4705	26244885	(47)
55	Predicting Variabilities in Cardiac Gene	4706	26207376	(48)
56	PC12 Cell Differentiation	4775	27148350	(49)
57	Human Gonadal Sex Determination	4779	26573569	(50)
58	IGVH mutations in chronic lymphocytic leukemia.	4783	26088082	(51)
59	Fanconi anemia and checkpoint recovery	4790	26385365	(52)
60	Arabidopsis thaliana Cell Cycle	4837	26340681	(53)
61	Bortezomib Responses in U266 Human Myeloma Cells	4850	26163548	(54)
62	Stomatal Opening Model	4932	27542373	(55)
63	Pro-inflammatory Tumor Microenvironment in Acute Lymphoblastic Leukemia	4942	27594840	(56)
64	CD4+ T Cell Differentiation and Plasticity	5025	26090929	(26)
65	Lac Operon	5128	21563979	(57)
66	Metabolic Interactions in the Gut Microbiome	5731	26102287	(58)
67	Tumour Cell Invasion and Migration	5884	26528548	(59)
68	CD4+ T cell Differentiation	6678	22871178	(6)
69	Regulation of the L-arabinose operon of Escherichia coli.	6885	28639170	(60)
70	Aurora Kinase A in Neuroblastoma	7916	26616283	(61)
71	Iron acquisition and oxidative stress response in aspergillus fumigatus.	7926	25908096	(62)
72	MAPK Cancer Cell Fate Network	7984	24250280	(63)
73	Treatment of Castration-Resistant Prostate Cancer	8048	28361666	(64)
74	Lymphopoiesis Regulatory Network	8080	26408858	(65)
75	Lymphoid and myeloid cell specification and transdifferentiation	8186	28584084	(66)
76	T-LGL Survival Network 2011 Reduced Network	8227	22102804	(32)
77	Senescence Associated Secretory Phenotype	11863	29206223	(67)
78	Signaling Pathway for Butanol Production in Clostridium beijerinckii NRRL B-598	36604	30718562	(68)

where we observe very strong effects for some conditions and not others. For instance, imagine condition $x_1 = 0 \wedge x_2 = 1 \wedge x_3 = 1$ leads to uncertain results about the expression of x_4; say, x_4 is expressed in only 60% of the experiments for that condition. In contrast, all other conditions lead to very certain observations of expression or inhibition of x_4. With this information, modelers, who have to decide on a acceptable threshold for evidence, may chose the relationship as $x_4 = (x_1 \wedge x_2) \vee (x_1 \wedge x_2 \wedge x_3)$ (LUT in Figure 1 in main text with $f_4 \equiv 011 : 0$) rather than $x_4 = (x_1 \wedge x_2) \vee (x_2 \wedge x_3)$ (LUT with $f_4 \equiv 011 : 1$).

Notice that in network inference, modelers often consider each node's LUT condition independently, as they can result from different experimental evidence. This means that the resulting LUTs are not necessarily further checked for logical redundancy or even incoherence (tautologies and contradictions). For instance, the *Thaliana* model (Figure 4, main text) contains three fully redundant edges, which ultimately result from "subjective decisions given alternatives with equivalent results" (7) regarding the expression of the *LFY* (*Leafy*) and *TFL1* (*Terminal Flower 1*) proteins. Certainly, our methodology to quantify redundancy in automata networks can also serve as an additional logical check on these models to avoid and understand the existence of completely redundant interactions derived from incomplete experimental evidence or modeling decisions.

3. Effectiveness Gini

Given a vector of n values $[x_1, x_2, \ldots, x_n]$, the Gini coefficient is defined as:

$$G = \frac{\sum_{i,j} |x_i - x_j|}{2n^2\bar{x}} \qquad [5]$$

where $\bar{x}$ is the mean of the values. Note that, due to finite-size effects, the maximal Gini coefficient for automata of degree $k = 6$ is only 0.833, found when the vector is $[0, 0, 0, 0, 0, 1]$. As the number of samples increases, the maximal Gini coefficient approaches 1.

4. The effective graph and the spread of perturbations

Here we consider bit-flip perturbations to individual variables, defined as altering the logical state of the variable. In principle, the same framework can be used to study more elaborate classes of dynamic perturbations, including multi-variable bit flips or

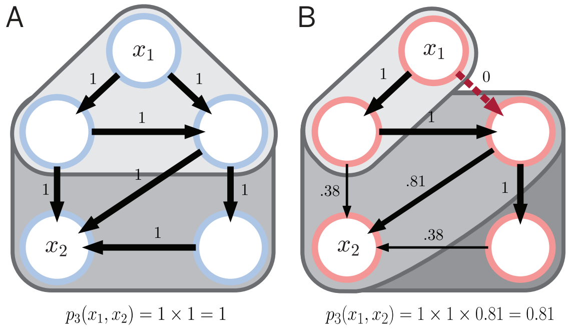

A
B
x_1
x_2
1
0
.81
.38
$p_3(x_1, x_2) = 1 \times 1 = 1$
$p_3(x_1, x_2) = 1 \times 1 \times 0.81 = 0.81$

Fig. S1. The effective graph captures the spread of perturbations. A In the $\mathcal{M}_{\text{IG}}$ model, perturbations from automata x_1 spread equally to all connected variables at t steps away in the interaction graph of an example BN. **B** In the $\mathcal{M}_{\text{EG}}$ model, the effective graph of a BN constrains the spread of perturbations.

pinning perturbations. Changes to the system structure by changing variable transition functions through the addition or removal of an input are not considered.

The impact of a perturbation on an automaton in a BN is quantified by the Boolean analogue of the partial derivative (71):

$$\partial_t^{(i)} x_j(\boldsymbol{x}_\alpha) = |x_j^t(\boldsymbol{x}_\alpha) - x_j^t(\boldsymbol{x}_\alpha^{\neg i})| \quad , \qquad [6]$$

where $x_j^t(\boldsymbol{x}_\alpha)$ denotes the state (truth value) of automaton node x_j at time t when the BN is initiated with configuration $\boldsymbol{x}^0 = \boldsymbol{x}_\alpha$ at time $t = 0$, and $\boldsymbol{x}_\alpha^{\neg i}$ denotes configuration $\boldsymbol{x}_\alpha$ with the state (truth value) of automaton x_i negated. In other words, the partial derivative yields 1 if flipping the state of x_i in initial configuration $\boldsymbol{x}^0$ leads to x_j flipping its state at time t, and 0 otherwise. The total impact on automaton x_j of perturbations to automaton x_i after t steps is found by averaging over all initial configurations:

$$\iota_{ij}(t) = 2^{-N} \sum_{\alpha=1}^{2^N} \partial_t^{(i)} x_j(\boldsymbol{x}_\alpha). \qquad [7]$$

For large BNs, the exact calculation of $\iota_{ij}(t)$ becomes computationally infeasible and is approximated by averaging over a random sample of initial network configurations.

The interaction graph defines the light-cone of perturbation spreading, but it cannot differentiate the potential impact to nodes within the cone. Specifically, since signals between two nodes cannot travel faster than the minimum number of edges (thus, time steps t) between them, the interaction graph provides an upper bound on the number of variables potentially affected by a perturbation after t time steps. We capture this upper bound in model $\mathcal{M}_{\text{IG}}$. The model $\mathcal{M}_{\text{IG}}$ partitions the nodes into two groups: all nodes connected via a path of at most t edges starting from node x_j are equally impacted by a perturbation to node x_j, where t is the number of time steps since the perturbation, and all other nodes are not impacted by the perturbation. For simplicity, this model considers only the minimum path length from the perturbed node x_j, and does not account for loops or multiple paths.

The effective graph has at least two advantages for capturing the spread of perturbations: 1) it more accurately defines the light-cone of perturbation spreading since it removes interactions that are fully redundant, and 2) it provide edge weights that can differentiate the potential impact to nodes within the cone. The second model $\mathcal{M}_{\text{EG}}$ thus ranks the node variables within the perturbation light-code based on the weights along the most effective path from x_j to the variable. Specifically, the propensity for a perturbation to be transmitted along a path in $\mathcal{M}_{\text{EG}}$ is given by the maximum product of edge strengths, constrained by the perturbation light-cone such that the number of edges in the path is less than the number of elapsed time steps. The maximum product path is calculated by finding the minimum additive path of negative log edge effectiveness.

Implementations of both models for perturbation spreading provided in the CANA python package(4).

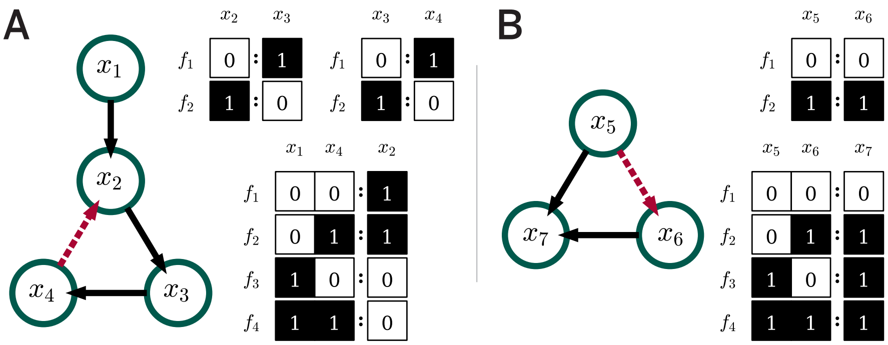
For the random BN experiment we generate 100 sample networks with $N = 100$ nodes, and in which all nodes have an in-degree of 3, and average bias $\bar{p} = 0.4$ (SI, S3). The network topology is randomized using a configuration model such that multi-edges are not allowed (each multi-edge is randomly swapped with another edge until no multi-edges remain). Self-loops are allowed. For each of the 100 randomly constructed networks, we sample 10 nodes at random to perturb. For each of the focus nodes, the dynamical impact on all other nodes is approximated using trajectories starting from 10^4 random initial configurations.

5. The effective graph improves structure-based control

The discovery of control strategies in BN models is a central problem in systems biology and biomedicine because predictions about controllability can help focus experimental interventions on genes, proteins and even medications more likely to result in the desired phenotype or medical outcome. Accurate control predictions would facilitate, for example, the design of advanced disease therapeutics(72, 73) or intervention strategies to reprogram cells(74), e.g. to revert a mutant cell to a wild-type state.

It is well known that when the set of automata nodes X of a BN is large, enumeration of all configurations $\boldsymbol{x} \in \mathcal{X}$ of its STG becomes difficult, making the controllability of BN an NP-hard problem (75). Therefore, control methodologies which leverage the interaction graph or otherwise simplify the dynamics are highly desirable since they can greatly simplify the complexity of BN control(3, 73).

Two recent methodologies aim to determine the controllability of complex dynamical systems based solely from the graph of interactions between variables: structural controllability (SC)(76), and feedback-vertex set control (FVC)(73, 77). By using only the structural graph to predict minimum sets of variables that are needed to control a network (a.k.a. driver nodes), both of these methods make predictions about the entire ensemble of dynamical systems that fit the same interaction graph—rather than a specific multivariate dynamical system, such as a BN(78).



	x_2	x_3
f_1	0	1
f_2	1	0

	x_3	x_4
f_1	0	1
f_2	1	0

	x_1	x_4	x_2
f_1	0	0	1
f_2	0	1	1
f_3	1	0	0
f_4	1	1	0

	x_5	x_6
f_1	0	0
f_2	1	1

	x_5	x_6	x_7
f_1	0	0	0
f_2	0	1	1
f_3	1	0	1
f_4	1	1	1

Fig. S2. Fully redundant edges in the effective graph and structural control. Two examples of BN in which canalization alters the predictions of structure-based control methods. **(A)**, A small BN with a fully canalized interaction between nodes x_4 and x_2 (dashed red). The predictions of SC and FVC on the structural graph suggest two nodes (x_1 and one of x_2, x_3, or x_4) are required to control the network dynamics, while the same methods applied to the effective graph accurately identify that only node x_1 is required to fully control the network. **(B)**, A small BN with a fully canalized interaction between nodes x_5 and x_6 (red). The predictions of SC, MDS and FVC on the structural graph suggest only node x_5 is required to control the network dynamics, while the same methods applied to the effective graph accurately identify that both nodes x_5 and x_6 are required to fully control the network.

Here, we demonstrate that the effective graph is a more accurate representation of interactions between variables with important consequences for structural approximations of control, such as SC and FVC. Specifically, the redundancy of some logical functions means that the effective structure of interactions is reduced: fully-canalized edges of the structural graph play no role in determining the transitions between configurations. The presence of such canalization can both decrease or increase the estimated driver variable sets using structure-based control methods.

We illustrate how canalization can reduce the number of predicted driver variables using the BN shown in Fig. S2**A**. Both SC and FVC would predict that interventions on two nodes are required to control the system dynamics (x_1 and one of x_2, x_3, or x_4.). However, the interaction between nodes x_4 and x_2 is fully canalized (red edge), meaning that it should be disregarded by the structure-based methods. Applying both SC and FVC to the effective graph correctly reveals that only node x_1 is required to control the system.

On the other hand, canalization can also increase the number of predicted driver variables as illustrated by the BN shown in Fig. S2**B**. In this case, both SC and FVC would predict that interventions on node x_5 are sufficient to control the system dynamics. The effective graph reveals that the interaction between nodes x_5 and x_6 is full canalized (red edge); using the effective graph, SC and FVC correctly predict that both nodes x_5 and x_6 are required to control the system.

6. Effective graph and control

Similar insights about the control patterns from individual driver variables can be seen when comparing the minimum driver variable set for pinning controllability predicted by FVC (73, 77), with the real one obtained by full enumeration of all possible node sets in the STG (78). Recall that pinning controllability specifies a system can be controlled from any initial configuration to any of its attractors via "pinning" the driver variables to their state(s) in the target attractor (77). For the *Arabidopsis thaliana* BN model, in addition to the input nodes, FVC predicts the network to be pinning controllable with interventions to the 6 additional variables $D_{FVC} = \{WUS, AP3, AG, TFL1, LFY, PI\}$ However, enumeration of the STG (78) reveals that there are actually 3 equivalent minimum driver variable sets required for pinning control, each with only 5 additional variables: $D_{pin} = \{WUS, AP3, AG\} \cup \{\{AP1, LFY\} \vee \{TFL1, EMF1\} \vee \{TFL1, LFY\}\}$. The effective graph reveals why FVC overestimates this minimum set, and why a multiplicity of sets are equally effective in pinning control. First, note that all of the self-loops to PI, $AP3$, and AG have negligible edge effectiveness, and thus do not does not need to be controlled because its self-loop is negligible and edges with stronger effectiveness exist from LFY, $AP3$ and AG that can control it (see Fig. 4**C**).

The effective graph reveals that some loops are removed entirely with fully redundant edges (i.e. $TFL1 \longleftrightarrow AP2$ and $AP1 \longleftrightarrow LFY$), or almost removed with very low effectiveness edges (e.g. the PI and AG self-loops), which help explain the differences between the FVC-predicted and real pinning control driver sets. Several interesting observations derive by looking at the effective graph and comparing D_{FVC} with D_{Pin}. Recall that FVC theory requires that all loops, including variable self-loops, need to be controlled by at least one variable that interrupts the loop. The fully redundant edge between $AP2$ and $TFL1$ removes the loop between these variables which means that $TFL1$ is not always necessary to control the network and $AP1$ with LFY can control $TFL1$ and $AP2$. Similarly, in the real pinning control driver set, $EMF1$ can take the place of LFY, which is not allowed by the FVC prediction. The effective graph shows that this happens because the fully redundant edge between $AP1$ and LFY removes the loop between these variables, and so LFY is not always needed to control the network—since all the loops with effective edges ($e_{ij} > 0$) in which LFY participates can be interrupted by pining $TFL1$ and $EMF1$.

These observations demonstrate that while FVC predicts the necessary driver set to control the entire ensemble of BN that fit the same interaction graph (Fig. 4**A**), the effective graph of a specific BN can reduce the size of the necessary driver set and help identify alternative control strategies and most important variables to control, as is the case of the effective graph of the TBN model (Fig. 4**B**). These features make the effective graph useful to analyze control propagation in BN systems biology models, by providing more specific understanding of the effective pathways to control dynamics. This suggests that applying FVC to the effective graph can lead to more accurate (pinning) control predictions. The development of such a method is beyond the scope of this article, but we can at least demonstrate that in the case of the TBN model the effective graph with a threshold of $e_{ij} >= 0.2$ (Fig. S4**)** best explains the real pinning control driver set. If we apply FVC to this graph the result is $D_{Pin} - AG$. That is, it only misses the AG variable, which is needed for pinning control. A closer inspection of the full effective graph (Fig. 4**B**) reveals that AG participates in three low-effectiveness loops[*] that are not interrupted by other variables in D_{Pin}. In contrast, PI which is not needed for real control (not in D_{Pin} though in D_{FVC}), participates in a single low-effectiveness loop (its own self-loop) that is not interrupted by other variables in D_{Pin}. This strongly suggests that the number of low-effectiveness loops may be cumulative and needs to be accounted for, a hypothesis we will address in future work.

Similarly, in the real pinning control driver set, $EMF1$ can take the place of LFY, which is not allowed by the FVC prediction. The effective graph shows that this happens because the fully redundant edge between $AP1$ and LFY removes the loop between these variables, and so LFY is not always needed to control the network—since all the loops with effective edges ($e_{ij} > 0$) in which LFY participates can be interrupted by pining $TFL1$ and $EMF1$.

In summary, we need to control LFY because it can control every other node, but we need to control WUS only for its own sake. This type of information is very useful for considering intervention strategies in Biology. If WUS's final state is not of high importance, we can control most of the network by intervening only on LFY: its impact/power on the rest of the network is much higher.

7. Effective graph reveals dynamically-decoupled modules

The analysis of perturbation spread in complex dynamical systems can reveal dynamical modules that constrain this spread (79). Such modules are often related to well-known pathways with important roles in biochemical regulation and signalling. As shown for the ER+ breast cancer and TBN models, by thresholding the effective graph and eliminating edges with small effectiveness, we reveal subgraphs with greater dynamical influence, as well as those that are more or less decoupled from the rest of the network. One way to characterize such dynamical modularity is to compute the strongly and weakly connected components of the effective graph for various effectiveness thresholds, and compare them to the interaction graph; Tables S2 & S3 in SI show such an analyses for the four networks studied in detail above. Strongly connected components reveal modules where every node can in principle perturb every other node in same module, and weakly connected components those where some of the (driver) nodes can perturb all nodes in module. The size of such components can also reveal how fractured the dynamics of a network is.

As shown in Fig. S23**A, B**, the 78 biochemical BN models from the Cell Collective vary in how dynamically-decoupled modules arise as we change the effectiveness threshold. For example, the TBN model (green) splits into several weakly connected components at a relatively low effectiveness value ($e_{ij} = 0.1$), but the largest component contains most of the network for a wide range of effectiveness values, at least 80% for $e_{ij} \leq 0.5$—which demonstrates the existence of the single primary dynamical module driven by the LFY protein described above. In contrast, the $ER+$ Breast Cancer model (orange) fractures into many small components at $e_{ij} \approx 0.4$, with the largest weakly connected component comprising less than 20% of the network—which is coherent with the patchwork composition of several dynamically distinct modules discussed above.

Overall, analysis of the 78 network models in the Cell Collective reveals that for edge effectiveness $e_{ij} \leq 0.2$ or even $e_{ij} \leq 0.4$, the majority of networks remain connected in a single or largest weakly connected component comprised of most nodes. As shown in Fig. S24**,** for $e_{ij} \leq 0.2$, about 90% of the networks have a largest weakly connected component comprised of at least 80% of the network. For $e_{ij} \geq 0.4$, on the other hand, most networks quickly lose a substantial largest weakly connected component, and for $e_{ij} \geq 0.65$, no networks have a largest weakly connected component comprised of even 70% of the network. That is, most networks break into many small components when $e_{ij} \in [0.4, 0, 6]$. In this sense, connectivity, signal transmission, and dynamical control in these networks tends to be robust to removal of edges with $e_{ij} \leq 0.2$ even up to $e_{ij} \leq 0.4$. This suggests $e_{ij} \in [0.2, 0, 4]$ is an optimum range for effectiveness, whereby redundant edges are removed but effective edges remain to reliably send signals through mostly connected networks.

[*] $AG \longleftrightarrow WUS$, and $AG \longleftrightarrow AP1$, $AG \longleftrightarrow AG$.

References

1. Quine WV (1955) A Way to Simplify Truth Functions. *American Mathematical Monthly* 62:627–631.
2. McCluskey EJ (1956) Minimization of boolean functions. *The Bell System Technical Journal* 35(6):1417–1444.
3. Marques-Pita M, Rocha LM (2013) Canalization and control in automata networks: body segmentation in Drosophila melanogaster. *PloS ONE* 8(3):e55946.
4. Correia RB, Gates AJ, Wang X, Rocha LM (2018) Cana: A python package for quantifying control and canalization in boolean networks. *Frontiers in Physiology* 9.
5. Shmulevich I, Kauffman SA (2004) Activities and sensitivities in boolean network models. *PRL* 93(4):048701.
6. Helikar T, et al. (2012) The cell collective: Toward an open and collaborative approach to systems biology. *BMC Systems Biology* 6:96.
7. Espinosa-Soto C, Padilla-Longoria P, Alvarez-Buylla ER (2004) A gene regulatory network model for cell-fate determination during Arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles. *The Plant Cell Online* 16(11):2923–2939.
8. Chaos Á, et al. (2006) From Genes to Flower Patterns and Evolution: Dynamic Models of Gene Regulatory Networks. *Journal of Plant Growth Regulation* 25(4):278–289.
9. Li F, Long T, Lu Y, Ouyang Q, Tang C (2004) The yeast cell-cycle network is robustly designed. *PNAS* 101:4781–4786.
10. Zañudo J, Scaltriti M, Albert R (2017) A network modeling approach to elucidate drug resistance mechanisms and predict combinatorial drug treatments in breast cancer. *Cancer Convergence* 1(1).
11. Helikar T, Konvalina J, Heidel J, Rogers JA (2008) Emergent decision-making in biological signal transduction networks. *Proceedings of the National Academy of Sciences* 105(6):1913–1918.
12. Raza S, et al. (2008) A logic-based diagram of signalling pathways central to macrophage activation. *BMC Systems Biology* p. 15.
13. Sahin Ö, et al. (2009) Modeling ERBB receptor-regulated G1/S transition to find novel targets for de novo trastuzumab resistance. *BMC Systems Biology* p. 20.
14. Rodríguez A, et al. (2012) A boolean network model of the FA/BRCA pathway. *Bioinformatics* 28(6):858–866.
15. Singh A, Nascimento JM, Kowar S, Busch H, Boerries M (2012) Boolean approach to signalling pathway modelling in hgf-induced keratinocyte migration. *Bioinformatics* 28(18):i495–i501.
16. Giacomantonio CE, Goodhill GJ (2010) A boolean model of the gene regulatory network underlying mammalian cortical area development. *PLoS Computational Biology* 6(9):13.
17. Calzone L, et al. (2010) Mathematical modelling of cell-fate decision in response to death receptor engagement. *PLoS Computational Biology* 6(3):15.
18. Kazemzadeh L, Cvijovic M, Petranovic D (2012) Boolean model of yeast apoptosis as a tool to study yeast and human apoptotic regulations. *Frontiers in physiology* 3:446.
19. Herrmann F, Groß A, Zhou D, Kestler HA (2012) A boolean model of the cardiac gene regulatory network determining first and second heart field identity. *PLOS ONE* 7(10):10.
20. Li S, Assmann SM (2006) Predicting essential components of signal transduction networks: A dynamic model of guard cell abscisic acid signaling. *PLoS Biology* 4(10):17.
21. Saez-Rodriguez J, et al. (2007) A logical model provides insights into t cell receptor signaling. *PLoS Computational Biology* 3(8):11.
22. Kervizic G, Corcos L (2008) Dynamical modeling of the cholesterol regulatory pathway with boolean networks. *BMC Systems Biology* p. 14.
23. Zhang R, et al. (2008) Network model of survival signaling in large granular lymphocyte leukemia. *Proceedings of the National Academy of Sciences* 105(42):16308–16313.
24. Gupta S, Bisht SS, Kukreti R, Jain S, Brahmachari SK (2007) Boolean network analysis of a neurotransmitter signaling pathway. *Journal of Theoretical Biology* p. 7.
25. Ryll A, Samaga R, Schaper F, Alexopoulos LG, Klamt S (2011) Large-scale network models of IL-1 and IL-6 signalling and their hepatocellular specificationw. *Molecular BioSystems* p. 18.
26. Martinez-Sanchez ME, Mendoza L, Villarreal C, Alvarez-Buylla ER (2015) A minimal regulatory network of extrinsic and intrinsic factors recovers observed patterns of CD4+ t cell differentiation and plasticity. *PLOS Computational Biology* p. 23.
27. Samaga R, Saez-Rodriguez J, Alexopoulos LG, Sorger PK (2009) The logic of EGFR/ErbB signaling: Theoretical properties and analysis of high-throughput data. *PLoS Computational Biology* 5(8):19.
28. Mai Z, Liu H (2009) Boolean network-based analysis of the apoptosis network: Irreversible apoptosis and stable surviving. *Journal of Theoretical Biology* p. 10.
29. Méndez A, Mendoza L (2016) A network model to describe the terminal differentiation of b cells. *PLOS Computational Biology* p. 26.
30. Fauré A, Naldi A, Chaouiya C, Thieffry D (2006) Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics* 22(14):e124–e131.
31. Todd RG (2012) Ergodic sets as cell phenotype of budding yeast cell cycle. *PLOS ONE* 7(10):10.
32. Saadatpour A, et al. (2011) Dynamical and structural analysis of a t cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia. *PLoS Comput Biol* 7(11):e1002267.

33. Irons DJ (2009) Logical analysis of the budding yeast cell cycle. *Journal of Theoretical Biology* p. 17.
34. Mbodj A, Junion G, Brun C, Furlong EEM, Thieffry D (2013) Logical modelling of drosophila signalling pathways. *Molecular BioSystems* 9(9):2248.
35. Orlando DA, et al. (2008) Global control of cell-cycle transcription by coupled CDK and network oscillators. *Nature* 453:5.
36. Klamt S, Saez-Rodriguez J, Lindquist JA, Simeoni L, Gilles ED (2006) A methodology for the structural and functional analysis of signaling and regulatory networks. *BMC Bioinformatics* p. 26.
37. der Heyde S, et al. (2014) Boolean ErbB network reconstructions and perturbation simulations reveal individual drug response in different breast cancer cell lines. *BMC Systems Biology* 8(1):75.
38. Oyeyemi OJ, Davies O, Robertson DL, Schwartz JM (2015) A logical model of hiv-1 interactions with the t-cell activation signalling pathway. *Bioinformatics* 31(7):1075–1083.
39. Mendoza L, Xenarios I (2006) A method for the generation of standardized qualitative dynamical systems of regulatory networks. *Theoretical Biology and Medical Modelling* p. 18.
40. Madrahimov A, Helikar T, Kowal B, Lu G, Rogers J (2013) Dynamics of influenza virus and human host interactions during infection and replication cycle. *Bulletin of Mathematical Biology* 75(6):988–1011.
41. Silva-Rocha R, de Lorenzo V (2013) The tol network of p seudomonas putida mt-2 processes multiple environmental inputs into a narrow response space. *Environmental microbiology* 15(1):271–286.
42. Thakar J, Pathak AK, Murphy L (2012) Network model of immune responses reveals key effectors to single and co-infection dynamics by a respiratory bacterium and a gastrointestinal helminth. *PLoS Computational Biology* 8(1):19.
43. Christensen TS, Oliveira AP, Nielsen J (2009) Reconstruction and logical modeling of glucose repression signaling pathways in saccharomyces cerevisiae. *BMC Systems Biology* p. 15.
44. Sridharan S, Layek R, Datta A, Venkatraj J (2012) Boolean modeling and fault diagnosis in oxidative stress response. *BMC Genomics* 13(Suppl 6):S4.
45. Conroy BD, et al. (2014) Design, assessment, and in vivo evaluation of a computational model illustrating the role of cav1 in cd4+ t-lymphocytes. *Frontiers in immunology* 5:599.
46. Lu J, et al. (2015) Network modelling reveals the mechanism underlying colitis-associated colon cancer and identifies novel combinatorial anti-cancer targets. *Scientific reports* 5:14739.
47. Chasapi A, et al. (2015) An extended, boolean model of the septation initiation network in s.pombe provides insights into its regulation. *PLOS ONE* p. 22.
48. Grieb M, et al. (2015) Predicting variabilities in cardiac gene expression with a boolean network incorporating uncertainty. *PLOS ONE* p. 15.
49. Offermann B, et al. (2016) Boolean modeling reveals the necessity of transcriptional regulation for bistability in pc12 cell differentiation. *Frontiers in genetics* 7:44.
50. Ríos O, et al. (2015) A boolean network model of human gonadal sex determination. *Theoretical Biology and Medical Modelling* 12(1):1–18.
51. Álvarez Silva MC (2015) Proteins interaction network and modeling of IGVH mutational status in chronic lymphocytic leukemia. *Theoretical Biology and Medical Modelling* p. 15.
52. Rodríguez A (2015) Fanconi anemia cells with unrepaired DNA damage activate components of the checkpoint recovery process. *Theoretical Biology and Medical Modelling* p. 22.
53. Ortiz-Gutiérrez E, García-Cruz K, Azpeitia E, Castillo A (2015) A dynamic gene regulatory network model that recovers the cyclic behavior of arabidopsis thaliana cell cycle. *PLOS Computational Biology* p. 28.
54. Chudasama VL, Ovacik MA, Abernethy DR, Mager DE (2015) Logic-based and cellular pharmacodynamic modeling of bortezomib responses in u266 human myeloma cells. *Journal of Pharmacology and Experimental Therapeutics* 354(3):448–458.
55. Gan X (2016) Analysis of a dynamic model of guard cell signaling reveals the stability of signal propagation. *BMC Systems Biology* p. 14.
56. Enciso J, Mayani H, Mendoza L, Pelayo R (2016) Modeling the pro-inflammatory tumor microenvironment in acute lymphoblastic leukemia predicts a breakdown of hematopoietic-mesenchymal communication networks. *Frontiers in physiology* 7:349.
57. Veliz-Cuba A, Stigler B (2011) Boolean models can explain bistability in the lac operon. *Journal of computational biology* 18(6):783–794.
58. Steinway SN, Biggs MB, Loughran TP, Papin JA, Albert R (2015) Inference of network dynamics and metabolic interactions in the gut microbiome. *PLOS Computational Biology* p. 25.
59. Cohen DPA, et al. (2015) Mathematical modelling of molecular pathways enabling tumour cell invasion and migration. *PLOS Computational Biology* p. 29.
60. Jenkins A, Macauley M (2017) Bistability and asynchrony in a boolean model of the l-arabinose operon in escherichia coli. *Bulletin of mathematical biology* 79(8):1778–1795.
61. Dahlhaus M (2016) Boolean modeling identifies greatwall/MASTL as an important regulator in the AURKA network of neuroblastoma. *Cancer Letters* p. 11.
62. Brandon M, Howard B, Lawrence C, Laubenbacher R (2015) Iron acquisition and oxidative stress response in aspergillus fumigatus. *BMC Systems Biology* p. 18.
63. Grieco L, Calzone L, Bernard-Pierrot I (2013) Integrative modelling of the influence of MAPK network on cancer cell fate

decision. *PLOS Computational Biology* 9(10):15.

64. Arshad OA (2017) Towards targeted combinatorial therapy design for the treatment of castration-resistant prostate cancer. *BMC Bioinformatics* p. 11.
65. Mendoza L (2015) A dynamical model of the regulatory network controlling lymphopoiesis. *Biosystems* p. 8.
66. Collombet S, et al. (2017) Logical modeling of lymphoid and myeloid cell specification and transdifferentiation. *Proceedings of the National Academy of Sciences* 114(23):5792–5799.
67. Meyer P, et al. (2017) A model of the onset of the senescence associated secretory phenotype after dna damage induced senescence. *PLoS computational biology* 13(12):e1005741.
68. Patakova P, et al. (2019) Acidogenesis, solventogenesis, metabolic stress response and life cycle changes in clostridium beijerinckii nrrl b-598 at the transcriptomic level. *Scientific reports* 9(1):1–21.
69. Bitbol AF (2018) Inferring interaction partners from protein sequences using mutual information. *PLoS Computational Biology* 14(11):e1006401.
70. James R, Crutchfield J (2017) Multivariate dependence beyond shannon information. *Entropy* 19(10):531.
71. Luque B, Solé RV (2000) Lyapunov exponents in random Boolean networks. *Physica A: Statistical Mechanics and its Applications* 284(1-4):33–45.
72. Zhang R, et al. (2008) Network model of survival signaling in large granular lymphocyte leukemia. *PNAS* 105:16308–16313.
73. Zanudo JGT, Yang G, Albert R (2017) Structure-based control of complex networks with nonlinear dynamics. *PNAS* 114(28):7234–7239.
74. Wang RS, Albert R (2011) Elementary signaling modes predict the essentiality of signal transduction network components. *BMC Systems Biology* 5.
75. Akutsu T, Hayashida M, Ching WK, Ng MK (2007) Control of Boolean networks: hardness results and algorithms for tree structured networks. *Journal of Theoretical Biology* 244(4):670–679.
76. Liu YY, Slotine JJ, Barabási AL (2011) Controllability of complex networks. *Nature* 473:167–173.
77. Fiedler B, Mochizuki A, Kurosawa G, Saito D (2013) Dynamics and control at feedback vertex sets. i: Informative and determining nodes in regulatory networks. *Journal of Dynamics and Differential Equations* 25(3):563–604.
78. Gates AJ, Rocha LM (2016) Control of complex networks requires both structure and dynamics. *Scientific Reports* 6:24456.
79. Kolchinsky A, Gates AJ, Rocha LM (2015) Modularity and the spread of perturbations in complex dynamical systems. *Phy. Rev. E* 92(6):060801.

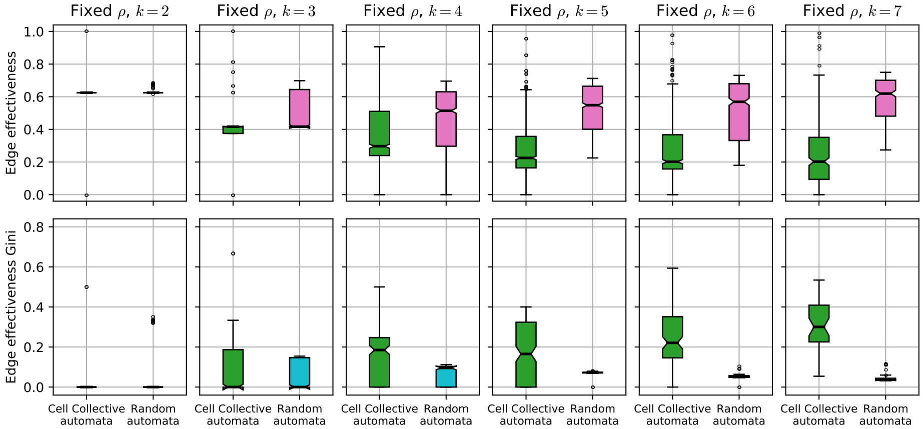


Fig. S3. Edge effectiveness of Boolean automata in biochemical regulation and random ensembles of varying degree. Top) Edge effectiveness of the 240 incoming edges (interactions) to 40 automata with varying degree in Cell Collective models (green) compared to a bias-matched sample of random Boolean automata (pink). Bottom) The Gini coefficient of the effectiveness of the incoming edges to the 40 automata with varying degree (green) in the Cell Collective models compared to the bias-matched ensemble of random Boolean automata (blue).

	Arabidopsis thaliana			
	IG	EG	EG 0.2	EG 0.4
nodes	15			
self-loops	5 (33%)			
input nodes	3 (20%)	3 (20%)	3 (20%)	4 (27%)
weakly conn. comp.	1 (15, 100%)	1 (15, 100%)	3 (13, 87%)	4 (12, 80%)
‘ strongly conn. comp.	5 (10, 67%)	6 (9, 60%)	8 (7, 47%)	10 (6, 40%)

	Saccharomyces cerevisiae (yeast)			
	IG	EG	EG 0.2	EG 0.4
nodes	12			
self-loops	8 (67%)			
input nodes	1 (8%)	1 (8%)	1 (8%)	1 (8%)
weakly conn. comp.	1 (12, 100%)	1 (12, 100%)	1 (12, 100%)	1 (12, 100%)
strongly conn. comp.	3 (10, 83%)	3 (10, 83%)	3 (10, 83%)	3 (10, 83%)

	Leukemia			
	IG	EG	EG 0.2	EG 0.4
nodes	60			
self-loops	11 (18%)			
input nodes	6 (10%)	6 (10%)	6 (10%)	10 (17%)
weakly conn. comp.	1 (60, 100%)	1 (60, 100%)	1 (60, 100%)	2 (58, 97%)
strongly conn. comp.	12 (48, 80%)	12 (48, 80%)	27 (29, 48%)	47 (9, 15%)

	Breast Cancer			
	IG	EG	EG 0.2	EG 0.4
nodes	80			
self-loops	23 (29%)			
input nodes	18 (23%)	18 (23%)	21 (26%)	29 (36%)
weakly conn. comp.	1 (80, 100%)	1 (80, 100%)	3 (78, 98%)	12 (52, 65%)
strongly conn. comp.	45 (24, 30%)	45 (24, 30%)	52 (17, 21%)	70 (3, 4%)

Table S2. Structural characteristics of four biochemical regulation networks. Number (and proportion) of nodes that are self-loops and inputs. Number of weakly and strongly connected components that exist for each graph; for each case, also shown in brackets is the number of nodes in largest component, followed by the proportion of network in largest component.

	Thaliana			
{size: number of comp.}	IG	EG	EG 0.2	EG 0.4
weakly conn. comp.	{15: 1}	{15: 1}	{13: 1, 1: 2}	{12: 1, 1: 3}
strongly conn. comp.	{10: 1, 2: 1, 1: 3}	{9: 1, 2: 1, 1: 4}	{7: 1, 2: 1, 1: 6}	{6: 1, 1: 9}

	Saccharomyces cerevisiae (yeast)			
	IG	EG	EG 0.2	EG 0.4
weakly conn. comp.	{12: 1}	{12: 1}	{12: 1}	{12: 1}
strongly conn. comp.	{10: 1, 1: 2}	{10: 1, 1: 2}	{10: 1, 1: 2}	{10: 1, 1: 2}

	Leukemia			
	IG	EG	EG 0.2	EG 0.4
weakly conn. comp.	{60: 1}	{60: 1}	{60: 1}	{58: 1, 2: 1}
strongly conn. comp.	{48: 1, 2: 1, 1: 10}	{48: 1, 2: 1, 1: 10}	{29: 1, 4: 1, 2: 2, 1: 23}	{9: 1, 4: 1, 2: 2, 1: 43}

	Breast Cancer			
	IG	EG	EG 0.2	EG 0.4
weakly conn. comp.	{80: 1}	{80: 1}	{78: 1, 1: 2}	{52: 1, 13: 1, 4: 1, 3: 1, 1: 8}
strongly conn. comp.	{24: 1, 8: 1, 2: 5, 1: 38}	{24: 1, 8: 1, 2: 5, 1: 38}	{17: 1, 8: 1, 2: 5, 1: 45}	{3: 1, 2: 8, 1: 61}

Table S3. Size and Number of structural components of four biochemical regulation models. For each type, components are listed in decreasing order of size (number of nodes), with the number of components of that size shown after each ':'.

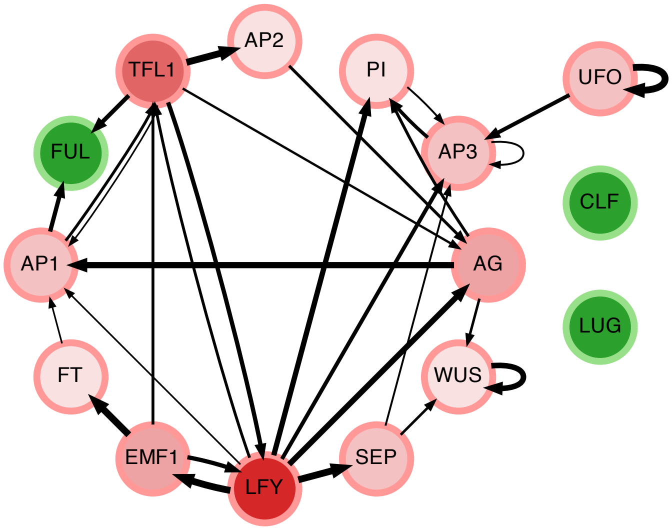


Fig. S4. The effective graph in the *Arabidopsis thaliana* BN model. The effective graph for the BN model of the *Arabidopsis thaliana*, in which edge thickness denotes its effectiveness, thresholded to $e_{ji} > 0.2$; node color intensity denotes the node effective out-degree; green nodes denote variables with no effective out-degree.

x_i	k	k_r	k_e	k_r^*	k_e^*	k^{out}	k_e^{out}	k_e^{out}/k^{out}
AG	9	6.9	2.1	0.77	0.23	5	1.9	0.38
AP3	7	4.7	2.3	0.68	0.32	2	0.8	0.4
PI	6	3.8	2.2	0.64	0.36	2	0.47	0.24
AP1	4	2.4	1.6	0.59	0.41	6	1.4	0.23
LFY	4	2.8	1.2	0.69	0.31	7	4.8	0.69
TFL1	4	2.8	1.2	0.69	0.31	5	2.8	0.57
WUS	3	1.4	1.6	0.48	0.52	2	0.91	0.46
FUL	2	0.75	1.2	0.38	0.62	1	0	0
UFO	1	0	1	0	1	2	1.6	0.79
FT	1	0	1	0	1	1	0.24	0.24
EMF1	1	0	1	0	1	3	2	0.68
AP2	1	0	1	0	1	2	0.43	0.22
SEP	1	0	1	0	1	4	0.9	0.22
LUG	0	0	1	0	0	1	0.1	0.1
CLF	0	0	1	0	0	1	0.1	0.1

Table S4. Canalization measures for variables in the *Arabidopsis thaliana* model. k, k_r, and k_e denote in-degree, input redundancy and effective connectivity, respectively; k_r^* and k_e^* denote versions of k_r, and k_e normalized by k; k^{out} and k_e^{out} denote out-degree and effective out-degree, respectively. Nodes with $k = 1$ have no redundancy ($k_r = 0, k_e = 1$), and input nodes have no incoming edges.

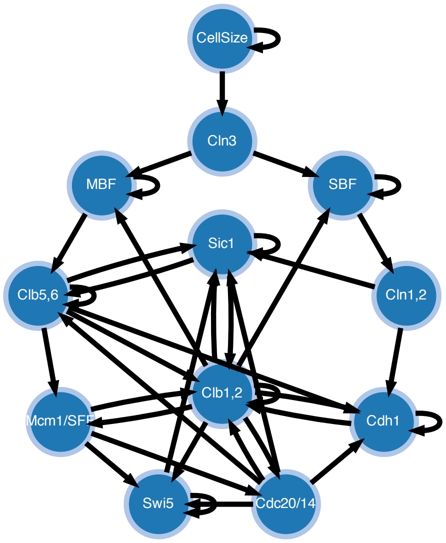


Fig. S5. The interaction graph in the *Saccharomyces cerevisiae* (yeast) BN model. The interaction graph for the BN model of *Saccharomyces cerevisiae*.

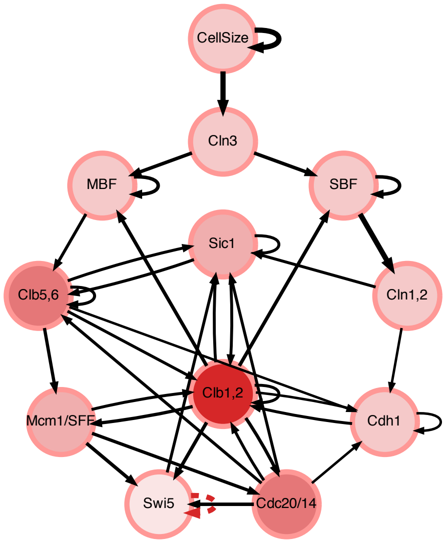


Fig. S6. The effective graph in the *Saccharomyces cerevisiae* (yeast) BN model. The effective graph for the BN model of *Saccharomyces cerevisiae*, in which edge thickness denotes its effectiveness, e_{ji}. Notice the fully redundant self-loop (dashed red) edge on the $Swi5$ transcription factor node.

Table S5. Canalization Measures for Variables in the *Saccharomyces cerevisiae* (yeast) model. k, k_r, and k_e denote in-degree, input redundancy and effective connectivity, respectively; k_r^* and k_e^* denote versions of k_r, and k_e normalized by k; k^{out} and k_e^{out} denote out-degree and effective out-degree, respectively. Nodes with $k = 1$ have no redundancy ($k_r = 0, k_e = 1$), and input nodes have no incoming edges.

x_i	k	k_r	k_e	k_r^*	k_e^*	k^{out}	k_e^{out}	k_e^{out}/k^{out}
Sic1	6	2.7	3.3	0.44	0.56	3	1.7	0.56
Clb1,2	6	2.7	3.3	0.44	0.56	8	4.8	0.6
Cdh1	5	2.6	2.4	0.53	0.47	2	1	0.52
Clb5,6	4	1.7	2.3	0.42	0.58	5	2.8	0.56
Swi5	4	2	2	0.5	0.5	2	0.56	0.28
SBF	3	1	2	0.33	0.67	2	1.7	0.83
MBF	3	1	2	0.33	0.67	2	1.2	0.62
Mcm1/SFF	2	0.75	1.2	0.38	0.62	3	1.8	0.62
Cdc20/14	2	0.75	1.2	0.38	0.62	5	2.8	0.57
CellSize	1	0	1	0	1	2	2	1
Cln3	1	0	1	0	1	2	1.3	0.67
Cln1,2	1	0	1	0	1	2	1	0.52

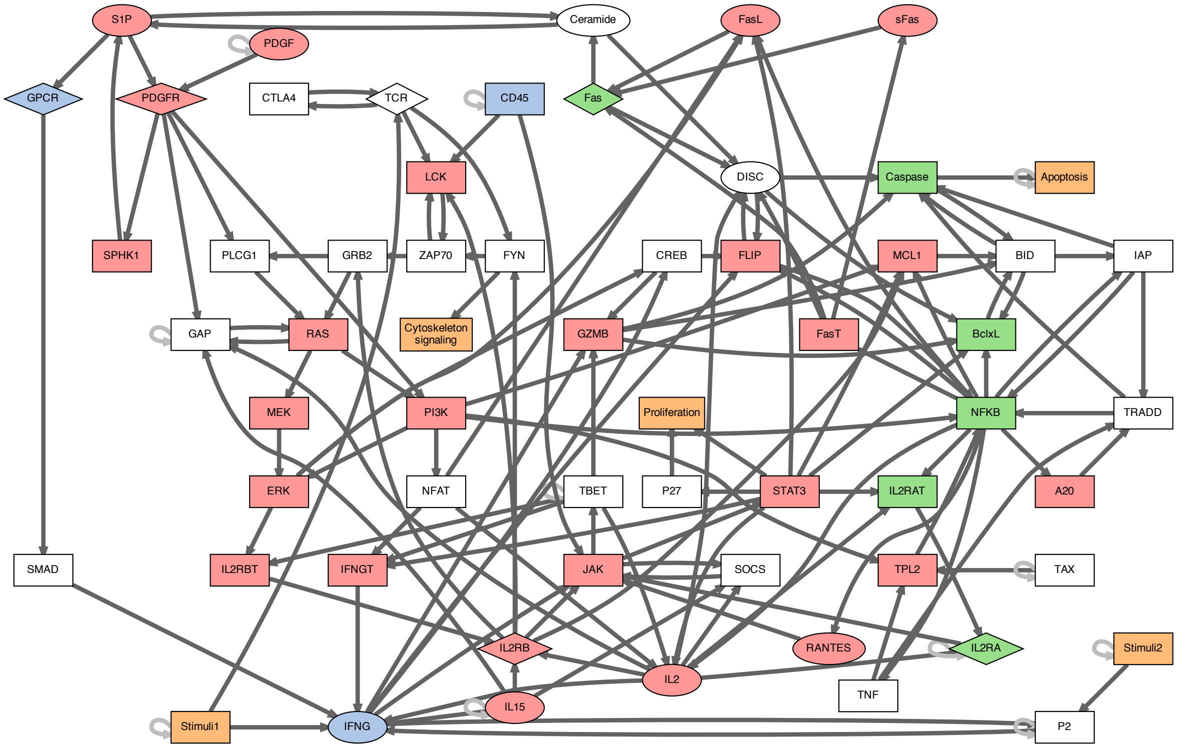


Fig. S7. The interaction graph in the Leukemia BN model. The interaction graph for the BN model of leukemia.

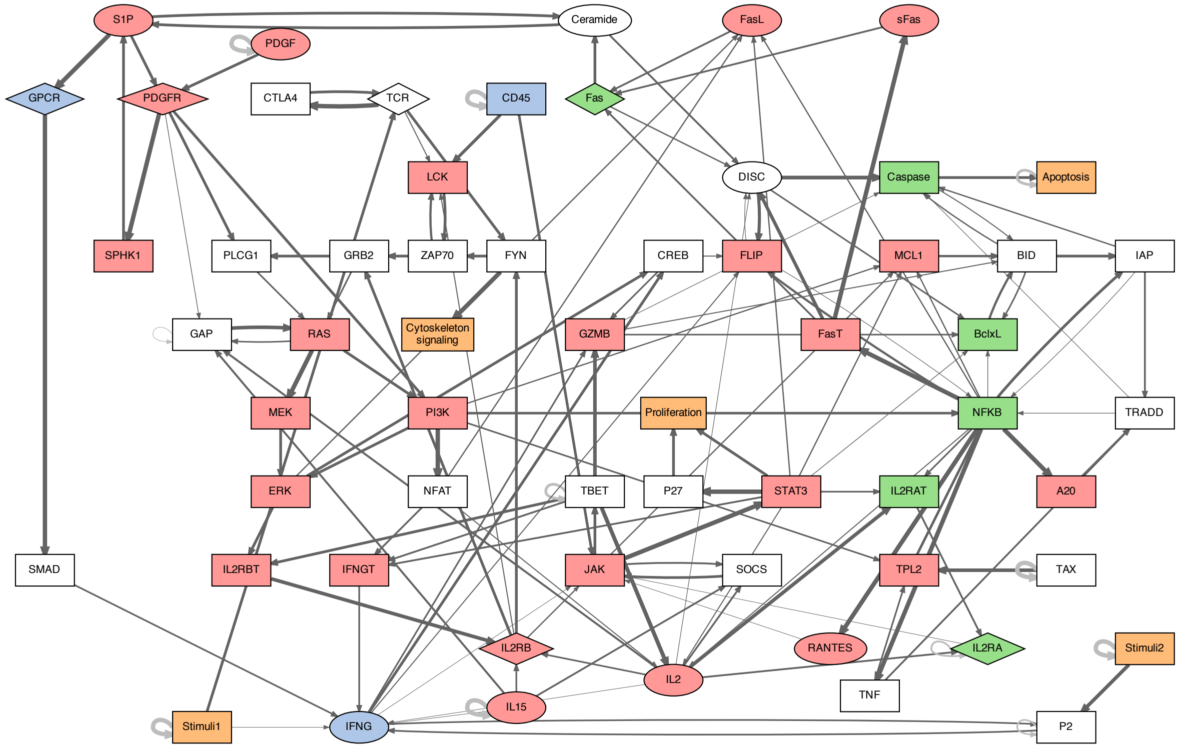


Fig. S8. The effective graph in the Leukemia BN model. The effective graph for the BN model of leukemia, in which edge thickness denotes its effectiveness, e_{ji}.

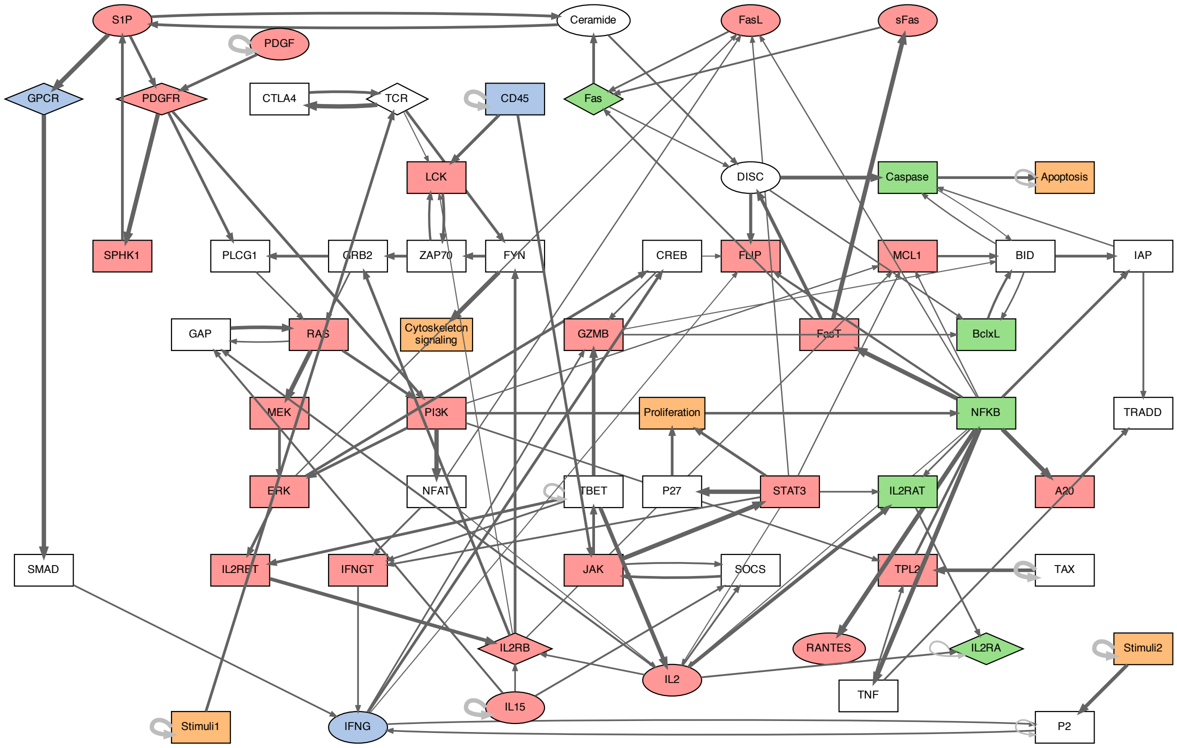


Fig. S9. The effective graph (thresholded to 0.2 in the Leukemia BN model. The effective graph for the BN model of leukemia, in which edge thickness denotes its effectiveness, thresholded to $e_{ji} > 0.2$.

Table S6. Canalization measures for variables in the *Leukemia* model. k, k_r, and k_e denote in-degree, input redundancy and effective connectivity, respectively; k_r^* and k_e^* denote versions of k_r, and k_e normalized by k; k^{out} and k_e^{out} denote out-degree and effective out-degree, respectively. Nodes with $k = 1$ have no redundancy ($k_r = 0, k_e = 1$), and input nodes have no incoming edges.

x_i	k	k_r	k_e	k_r^*	k_e^*	k^{out}	k_e^{out}	k_e^{out}/k^{out}
IFNG	6	4.6	1.4	0.76	0.24	5	1.7	0.34
JAK	6	4.4	1.6	0.74	0.26	3	2	0.68
GAP	5	3.4	1.6	0.68	0.32	2	0.98	0.49
Caspase	5	3.2	1.8	0.65	0.35	2	0.86	0.43
DISC	5	3.2	1.8	0.64	0.36	3	1.9	0.64
NFKB	5	3.4	1.6	0.68	0.32	11	6.5	0.59
BclxL	5	3.6	1.4	0.72	0.28	1	0.52	0.52
BID	4	2.5	1.5	0.62	0.38	3	1.3	0.43
FLIP	4	2.3	1.7	0.58	0.42	2	0.29	0.15
LCK	4	2.3	1.7	0.58	0.42	1	0.62	0.62
MCL1	4	2.8	1.2	0.7	0.3	1	0.52	0.52
FasL	4	2.8	1.2	0.7	0.3	1	0.42	0.42
IL2	4	2.4	1.6	0.59	0.41	7	2.7	0.39
IFNGT	3	1.8	1.2	0.58	0.42	1	0.39	0.39
IL2RB	3	1.4	1.6	0.48	0.52	5	1.9	0.38
Fas	3	1.7	1.3	0.58	0.42	2	0.94	0.47
IL2RA	3	1.7	1.3	0.58	0.42	2	0.51	0.26
IL2RAT	3	1.4	1.6	0.48	0.52	1	0.42	0.42
TRADD	3	1.7	1.3	0.58	0.42	2	0.29	0.15
P2	3	1.4	1.6	0.48	0.52	2	0.76	0.38
TPL2	3	1.4	1.6	0.48	0.52	1	0.57	0.57
GZMB	3	1.4	1.6	0.48	0.52	3	0.74	0.25
RAS	3	1.4	1.6	0.48	0.52	3	2	0.66
SOCS	3	1.7	1.2	0.58	0.42	1	0.6	0.6
GRB2	2	0.75	1.2	0.38	0.62	2	1	0.5
FYN	2	0.75	1.2	0.38	0.62	2	1.6	0.81
PDGFR	2	0.75	1.2	0.38	0.62	4	2.4	0.6
Apoptosis	2	0.75	1.2	0.38	0.62	1	0.62	0.62
IAP	2	0.75	1.2	0.38	0.62	3	0.88	0.29
Ceramide	2	0.75	1.2	0.38	0.62	2	1.1	0.53
TBET	2	0.75	1.2	0.38	0.62	5	3.4	0.68
TCR	2	0.75	1.2	0.38	0.62	3	1.9	0.62
PLCG1	2	0.75	1.2	0.38	0.62	1	0.38	0.38
ZAP70	2	0.75	1.2	0.38	0.62	2	1.1	0.57
Proliferation	2	0.75	1.2	0.38	0.62	0	0	-
S1P	2	0.75	1.2	0.38	0.62	3	2.2	0.75
ERK	2	0.75	1.2	0.38	0.62	3	1.5	0.52
CREB	2	0.75	1.2	0.38	0.62	2	0.61	0.31
IL2RBT	2	0.75	1.2	0.38	0.62	1	0.81	0.81
PI3K	2	0.75	1.2	0.38	0.62	5	2.9	0.57
Stimuli	1	0	1	0	1	3	1.7	0.57
Stimuli2	1	0	1	0	1	2	1.8	0.91
GPCR	1	0	1	0	1	1	1	1
IL15	1	0	1	0	1	5	2.3	0.47
CD45	1	0	1	0	1	3	2.3	0.77
SMAD	1	0	1	0	1	1	0.39	0.39
SPHK1	1	0	1	0	1	1	0.62	0.62
PDGF	1	0	1	0	1	2	1.6	0.81
CTLA4	1	0	1	0	1	1	0.62	0.62
A20	1	0	1	0	1	1	0.42	0.42
sFas	1	0	1	0	1	1	0.42	0.42
FasT	1	0	1	0	1	3	2.2	0.73
TNF	1	0	1	0	1	2	0.79	0.4
P27	1	0	1	0	1	1	0.62	0.62
STAT3	1	0	1	0	1	8	3.4	0.43
RANTES	1	0	1	0	1	1	0.095	0.095
NFAT	1	0	1	0	1	3	0.95	0.32
MEK	1	0	1	0	1	1	0.62	0.62
Cytoskeleton_signaling	1	0	1	0	1	0	0	-
TAX	1	0	1	0	1	2	1.8	0.91

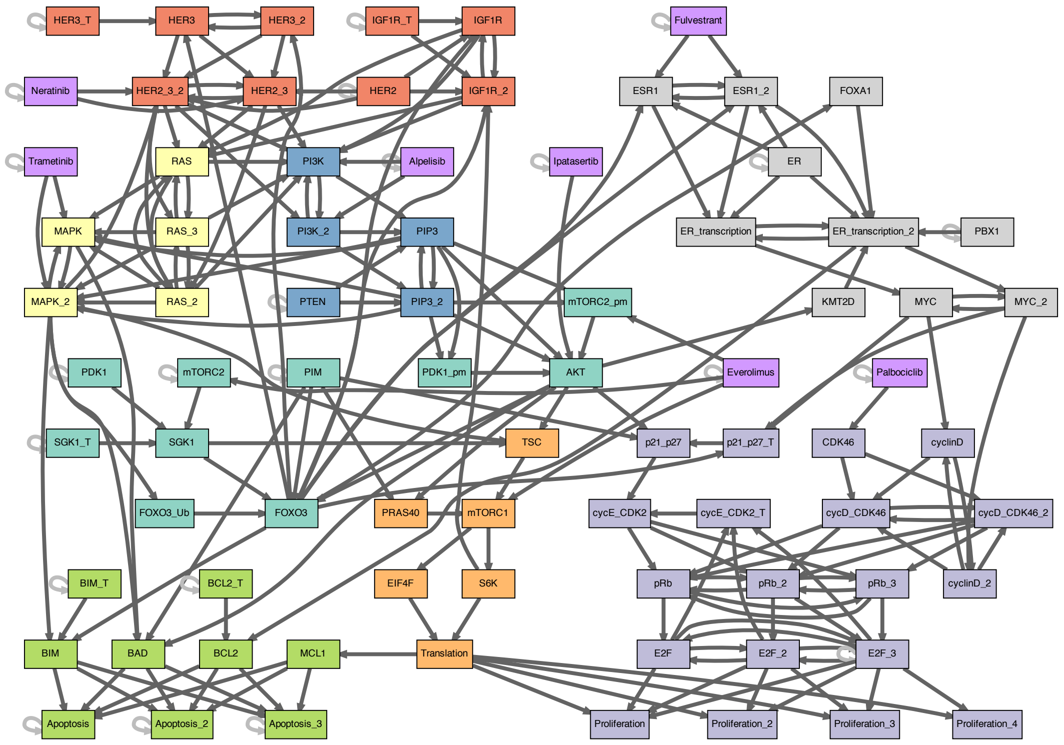


Fig. S10. The interaction graph in the ER+ breast cancer BN model. The goal of this model is to find interventions—especially single-node interventions—that synergize with the *PI3K* inhibitor $Alpelisib$, with particular interest on six other drugs used in cancer treatment: $Fulvestrant$, $Palbociclib$, $Everolimus$, $Neratinib$, $Trametinib$, and $Ipatersertib$ (all drug nodes in Purple). Specifically, the goal is to study how well these drugs control cancer cells to apoptosis or proliferation, which in this model are specific variables (10). This is done by running Monte-Carlo simulations of the BN while setting $Alpelisib$ to ON, in addition to setting baseline nodes to the cancerous state (Figs. S15-6), followed by setting the interventions to be tested to the appropriate state, for example, another Drug set to ON. See details in (10).

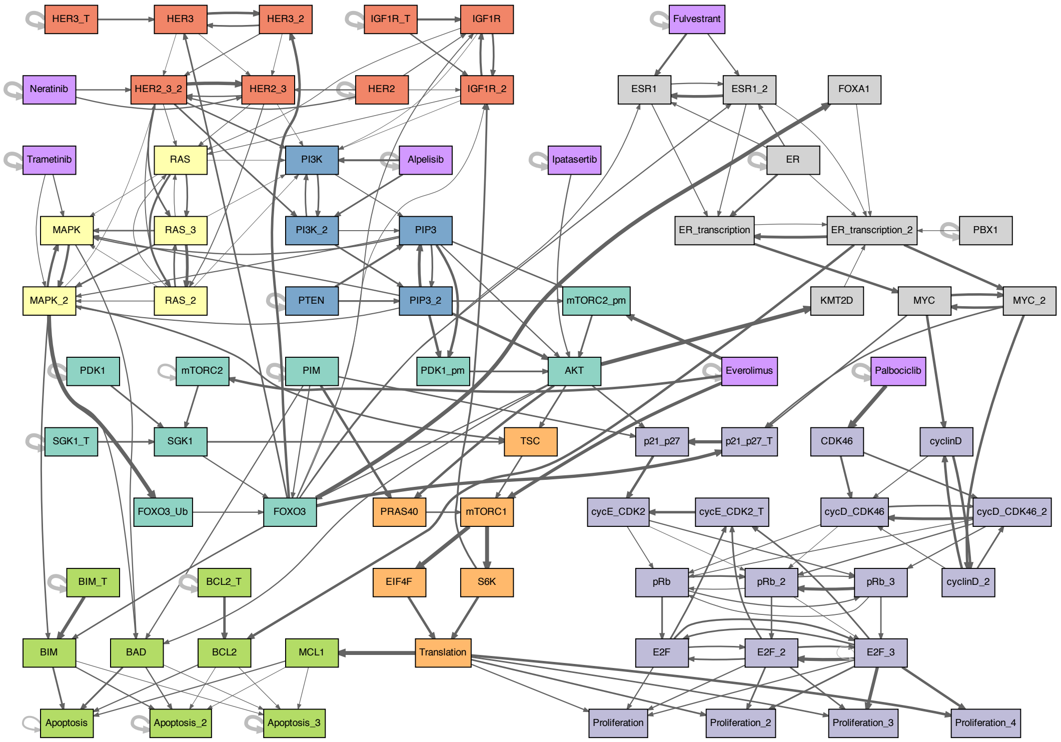


Fig. S11. The effective graph in the ER+ breast cancer BN model. Edge thickness denotes its effectiveness, e_{ji}.

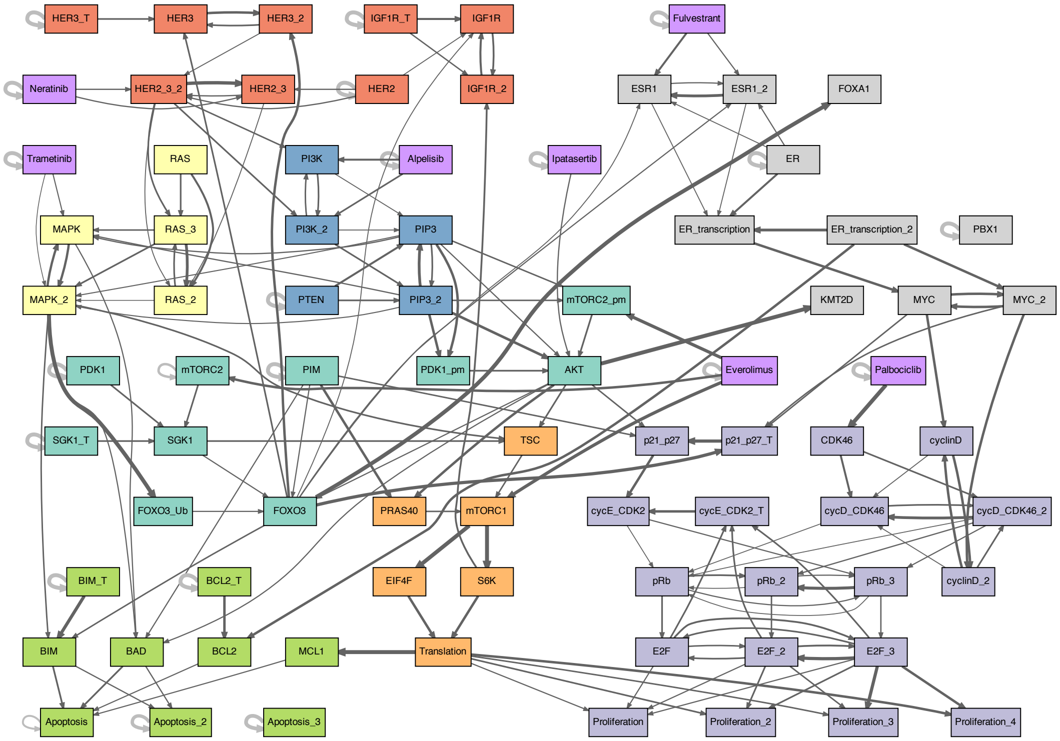


Fig. S12. The effective graph (thresholded to 0.2) in the ER+ breast cancer BN model. The effective graph for the BN model of ER+ breast cancer, in which edge thickness denotes its effectiveness, thresholded to $e_{ji} > 0.2$.

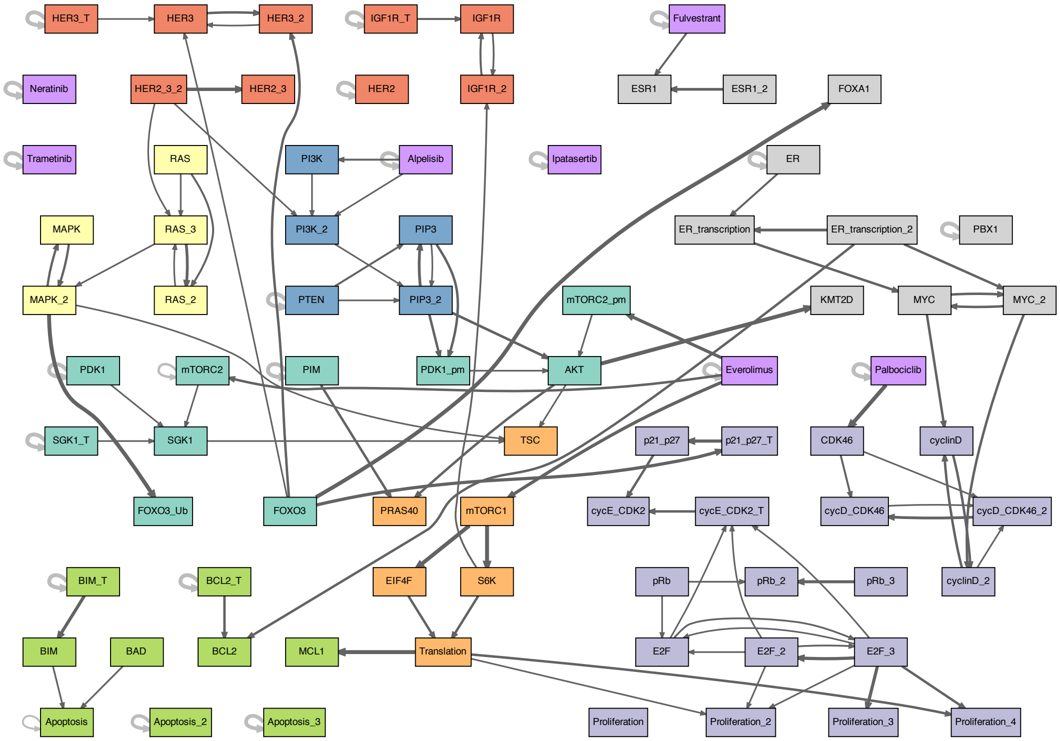


Fig. S13. The effective graph (thresholded to 0.4) in the ER+ breast cancer BN model. The effective graph for the BN model of ER+ breast cancer, in which edge thickness denotes its effectiveness, thresholded to $e_{ji} > 0.4$.

Table S7. Canalization measures for variables in the *Breast Cancer* model. k, k_r, and k_e denote in-degree, input redundancy and effective connectivity, respectively; k_r^* and k_e^* denote versions of k_r, and k_e normalized by k; k^{out} and k_e^{out} denote out-degree and effective out-degree, respectively. Nodes with $k = 1$ have no redundancy ($k_r = 0, k_e = 1$), and input nodes have no incoming edges.

x_i	k	k_r	k_e	k_r^*	k_e^*	k^{out}	k_e^{out}	k_e^{out}/k^{out}
PI3K	9	7.4	1.6	0.82	0.18	2	0.66	0.33
MAPK	7	4.9	2.1	0.7	0.3	2	0.88	0.44
MAPK_2	6	4	2	0.67	0.33	6	2.8	0.47
ER_transcription_2	6	4.9	1.1	0.82	0.18	3	2	0.65
RAS	6	4.9	1.1	0.82	0.18	4	1.1	0.28
HER2_3_2	6	4.5	1.5	0.76	0.24	6	2.5	0.41
pRb_2	5	3.2	1.8	0.64	0.36	4	1.1	0.27
AKT	5	2.9	2.1	0.57	0.43	6	3	0.5
HER2_3	5	3.2	1.8	0.65	0.35	4	0.81	0.2
pRb	5	3.9	1.1	0.78	0.22	3	1.2	0.38
Apoptosis_3	5	3.4	1.6	0.68	0.32	1	0.95	0.95
IGF1R_2	5	3.4	1.6	0.68	0.32	3	0.77	0.26
Apoptosis	5	3	2	0.61	0.39	1	0.49	0.49
E2F_3	5	3.4	1.6	0.68	0.32	8	4	0.5
Apoptosis_2	5	3.2	1.8	0.65	0.35	1	0.85	0.85
ER_transcription	4	2.3	1.7	0.58	0.42	2	0.8	0.4
PIP3	4	2.3	1.7	0.58	0.42	6	2.3	0.38
IGF1R	4	2.5	1.5	0.62	0.38	3	0.71	0.24
FOXO3	4	2.8	1.2	0.7	0.3	9	4.2	0.46
RAS_2	4	2.3	1.7	0.58	0.42	5	1	0.2
ESR1	4	2.3	1.7	0.58	0.42	2	0.54	0.27
Proliferation	4	2.8	1.2	0.7	0.3	0	0	-
cycD_CDK46	4	2.3	1.7	0.58	0.42	3	0.79	0.26
BAD	4	2.8	1.2	0.7	0.3	3	0.92	0.31
ESR1_2	4	2.8	1.2	0.7	0.3	3	1.1	0.38
pRb_3	4	2.8	1.2	0.7	0.3	3	1.3	0.44
BIM	3	1.4	1.6	0.48	0.52	3	0.92	0.31
mTORC1	3	1.4	1.6	0.48	0.52	2	2	1
TSC	3	1.8	1.2	0.58	0.42	1	0.38	0.38
cycE_CDK2_T	3	1.8	1.2	0.58	0.42	1	0.62	0.62
p21_p27	3	1.4	1.6	0.48	0.52	1	0.62	0.62
p21_p27_T	3	1.4	1.6	0.48	0.52	1	0.81	0.81
E2F	3	1.8	1.2	0.58	0.42	4	1.5	0.39
mTORC2_pm	3	1.4	1.6	0.48	0.52	1	0.43	0.43
RAS_3	3	1.7	1.3	0.58	0.42	5	1.8	0.35
E2F_2	3	1.4	1.6	0.48	0.52	6	2.4	0.4
cycD_CDK46_2	3	1.7	1.3	0.58	0.42	4	1.5	0.39
SGK1	3	1.7	1.3	0.58	0.42	2	0.71	0.36
Proliferation_2	3	1.8	1.2	0.58	0.42	0	0	-
Proliferation_3	3	1.4	1.6	0.48	0.52	0	0	-
PIP3_2	3	1.7	1.3	0.58	0.42	6	3	0.49
PI3K_2	3	1.7	1.3	0.58	0.42	3	1	0.34
HER3	3	1.8	1.2	0.58	0.42	3	0.89	0.3
cyclinD_2	2	0.75	1.2	0.38	0.62	3	1.3	0.43
Translation	2	0.75	1.2	0.38	0.62	5	2.7	0.54
MYC	2	0.75	1.2	0.38	0.62	3	1.6	0.54
MYC_2	2	0.75	1.2	0.38	0.62	3	1.6	0.54
cyclinD	2	0.75	1.2	0.38	0.62	2	0.86	0.43
BCL2	2	0.75	1.2	0.38	0.62	3	0.6	0.2
Proliferation_4	2	0.75	1.2	0.38	0.62	0	0	-
PRAS40	2	0.75	1.2	0.38	0.62	1	0.38	0.38
mTORC2	2	0.75	1.2	0.38	0.62	2	1	0.52
cycE_CDK2	2	0.75	1.2	0.38	0.62	3	0.67	0.22
PDK1_pm	2	0.75	1.2	0.38	0.62	1	0.43	0.43
HER3_2	2	0.75	1.2	0.38	0.62	3	0.81	0.27

Continues on the next page

Table S7 - *Continued from previous page*

x_i	k	k_r	k_e	k_r^*	k_e^*	k^{out}	k_e^{out}	k_e^{out}/k^{out}
S6K	1	0	1	0	1	2	1.1	0.54
Fulvestrant	1	0	1	0	1	3	1.8	0.6
Alpelisib	1	0	1	0	1	3	1.9	0.64
Everolimus	1	0	1	0	1	4	3.2	0.81
Trametinib	1	0	1	0	1	3	1.5	0.49
Ipatasertib	1	0	1	0	1	2	1.3	0.65
Palbociclib	1	0	1	0	1	2	2	1
Neratinib	1	0	1	0	1	3	1.6	0.54
HER2	1	0	1	0	1	5	2	0.41
HER3_T	1	0	1	0	1	2	1.4	0.71
PDK1	1	0	1	0	1	2	1.4	0.71
PIM	1	0	1	0	1	5	2.6	0.52
SGK1_T	1	0	1	0	1	2	1.4	0.71
ER	1	0	1	0	1	5	2.2	0.45
CDK46	1	0	1	0	1	2	0.93	0.46
PTEN	1	0	1	0	1	3	1.9	0.64
KMT2D	1	0	1	0	1	1	0.18	0.18
FOXO3_Ub	1	0	1	0	1	1	0.3	0.3
BIM_T	1	0	1	0	1	2	1.8	0.91
BCL2_T	1	0	1	0	1	2	1.6	0.81
PBX1	1	0	1	0	1	2	1.2	0.59
FOXA1	1	0	1	0	1	1	0.18	0.18
MCL1	1	0	1	0	1	3	0.6	0.2
EIF4F	1	0	1	0	1	1	0.62	0.62
IGF1R_T	1	0	1	0	1	3	1.9	0.62

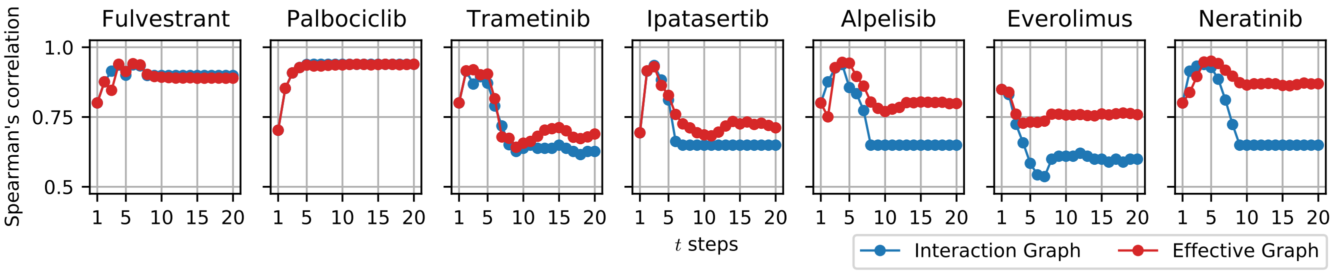


Fig. S14. The spread of perturbations in the ER+ breast cancer BN model. For each of the drug variable nodes, the predictive power of the path-length approximation using the interaction graph (blue), and the effective graph (red); measured by the Spearman's rank correlation (vertical axis) to the total impact of respective variable, after 20 steps (horizontal axis).

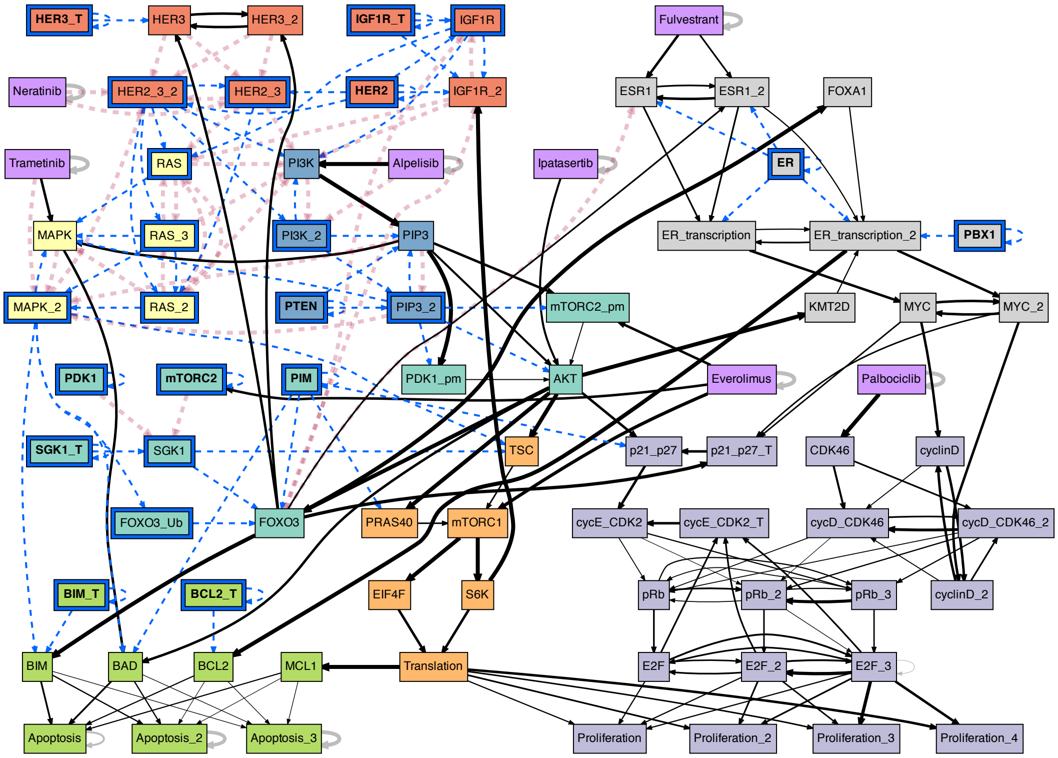


Fig. S15. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline. The baseline is defined by $K = \{ER =$ ON, $HER2 =$ OFF, $HER3\_T =$ OFF, $IGF1R\_T$ = ON, $PBX1 =$ ON, $PTEN =$ OFF, $SGK1\_T =$ OFF, $PIM1 =$ OFF, $PDK1\_T =$ OFF, $mTORC2 =$ OFF, $BIM\_T =$ OFF, and $BCL2\_T =$ OFF $\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that the $ER+/Her2-$ cancer cell state baseline alone resolves a substantial portion of the possible dynamics in comparison to the non-conditioned effective graph (Fig. S10). Strikingly, $Neratinib$ becomes redundant under this baseline initial condition, along with much of the HER pathway. Thus, $Neratinib$ has no effect on this model under $ER+/Her2-$ cancer cell state. Indeed, $Neratinib$ is one of the drugs that were shown not to synergize with $Alpelisib$ in (10); the others are $Ipasertib$ and $Trametinib$. We can see that these three drugs only contribute to the same pathways that $Alpelisib$ already acts on and become redundant when $Alpelisib$ is present (see Fig. S16, as well as Figs. S19, S20, S22).

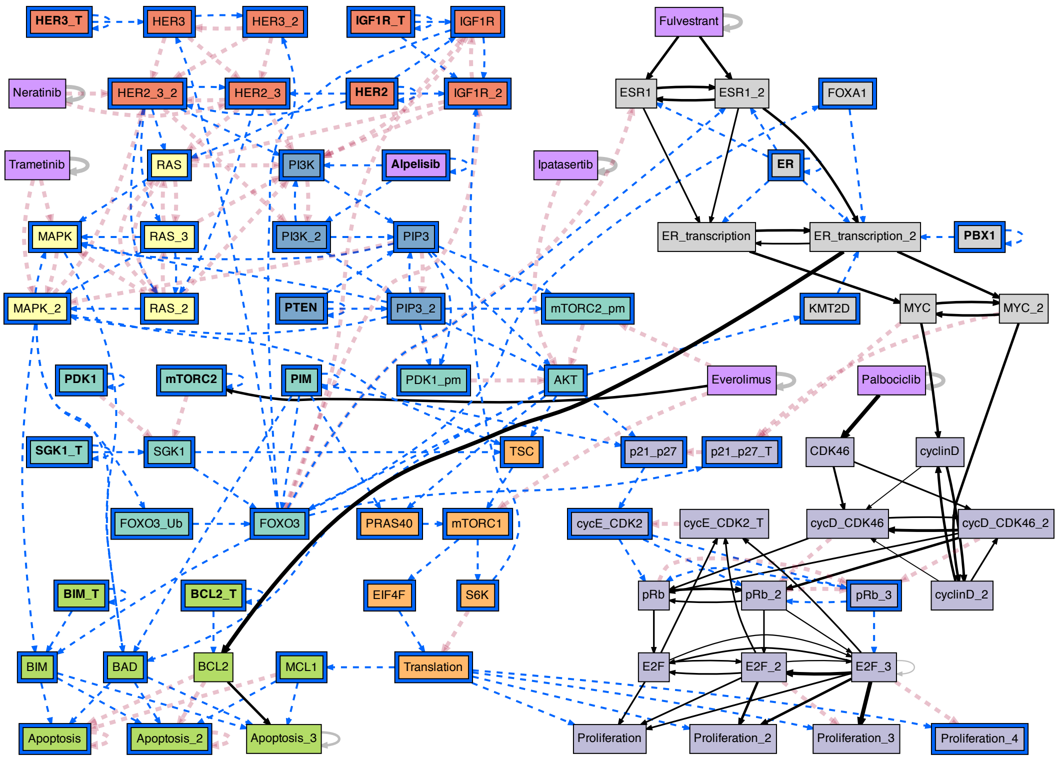


Fig. S16. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline resolves a majority of the possible dynamics in comparison to the non-conditioned effective graph (Fig. S10); unresolved dynamics is circumscribed almost entirely to ER and proliferation pathways. Interestingly, $Ipasertib$, $Neratinib$, and $Trametinib$ become redundant under this initial condition: they have no effect on model dynamics under $Alpelisib+$ $ER+/Her2-$ cancer cell state. The effective graph, however, reveals that the dynamics of this network is very robust to perturbation and hard to control *because* its subsystems are effectively decoupled. In particular, the baseline + *Alpelisib*=ON condition reveals that canalization works by preventing propagation of signals and cross-regulation. Indeed, most of the (non-drug) variables that have an impact on cancer apoptosis or proliferation under this condition (see Table 3 in (10)) have short paths to those target variables (at most 3 edges) in the effective graph. Exceptions in Table 3 in (10) are only a few nodes involved in the estrogen receptor (ER) transcription and signaling pathway such as the MYC oncogene and $KMT2D$ epigenetic transcription activator. Interestingly, the conditional effective graph reveals that $KMT2D$ becomes fixed under this condition, so any impact on proliferation can only occur by perturbing it out of its fixed state.

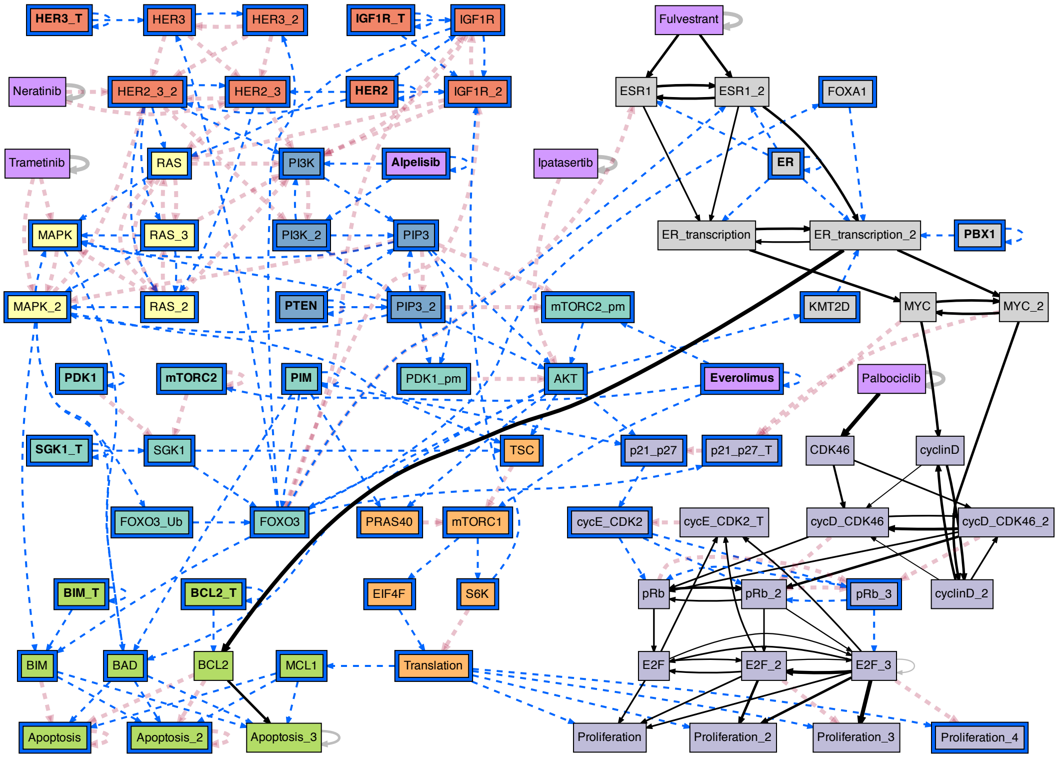


Fig. S17. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Everolimus*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Everolimus = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Everolimus$ hardly resolves any additional dynamics to what $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline already do (Fig. S16); only a few connections to AKT pathway.

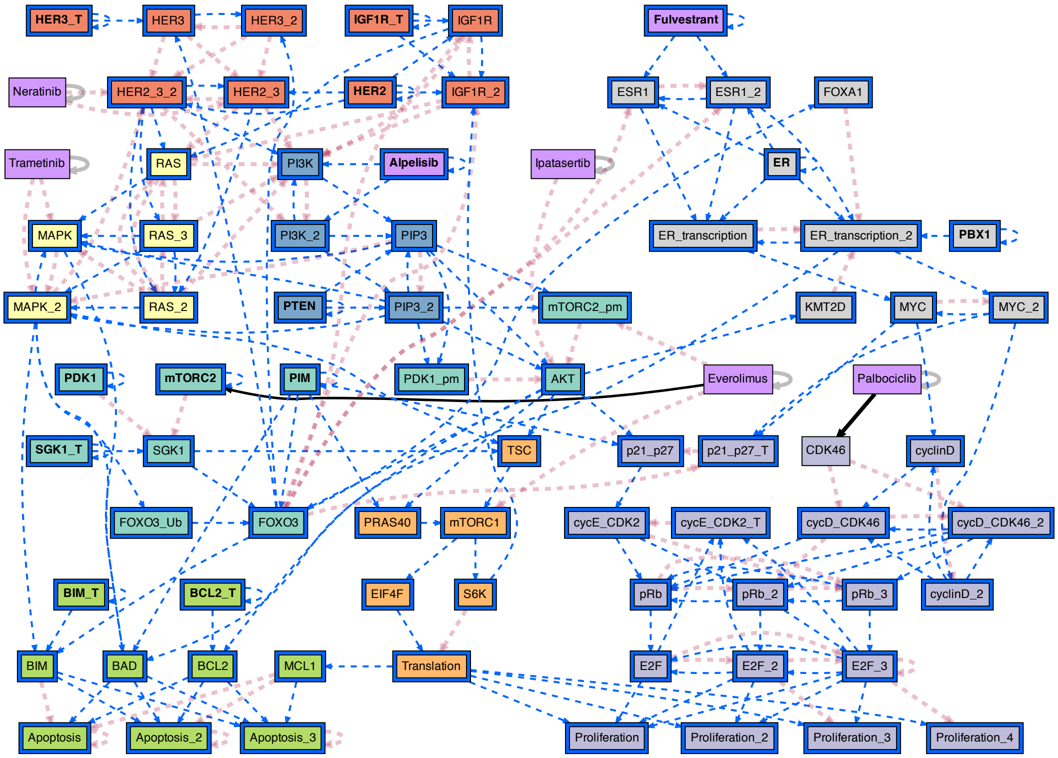


Fig. S18. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Fulvestrant*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Fulvestrant = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Fulvestrant$ resolves almost the remaining dynamics that was not yet resolved by $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline (Fig. S16. In particular, every apoptosis and proliferation node gets resolved. This combination strategy is the most powerful, rendering all other drugs redundant, except for the influence $Everolimus$ has in AKT pathway, which does not propagate anyway.

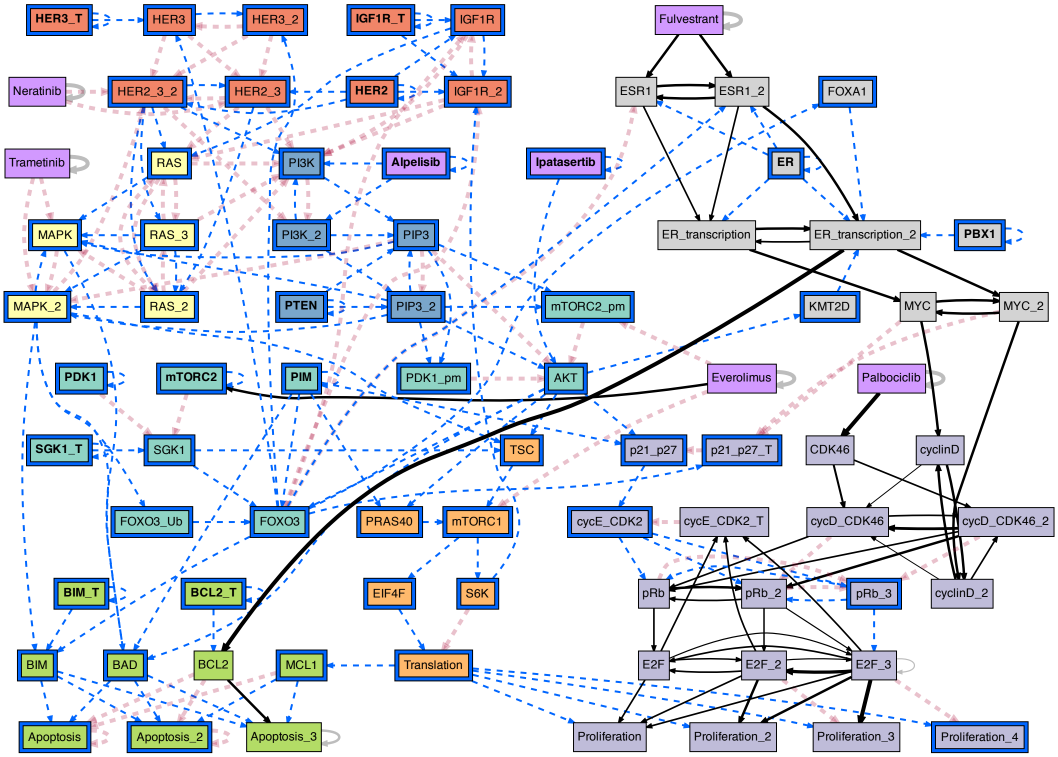


Fig. S19. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Ipatasertib*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Ipatasertib = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Ipatasertib$ does resolve any additional dynamics to what $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline already do (Fig. S16).

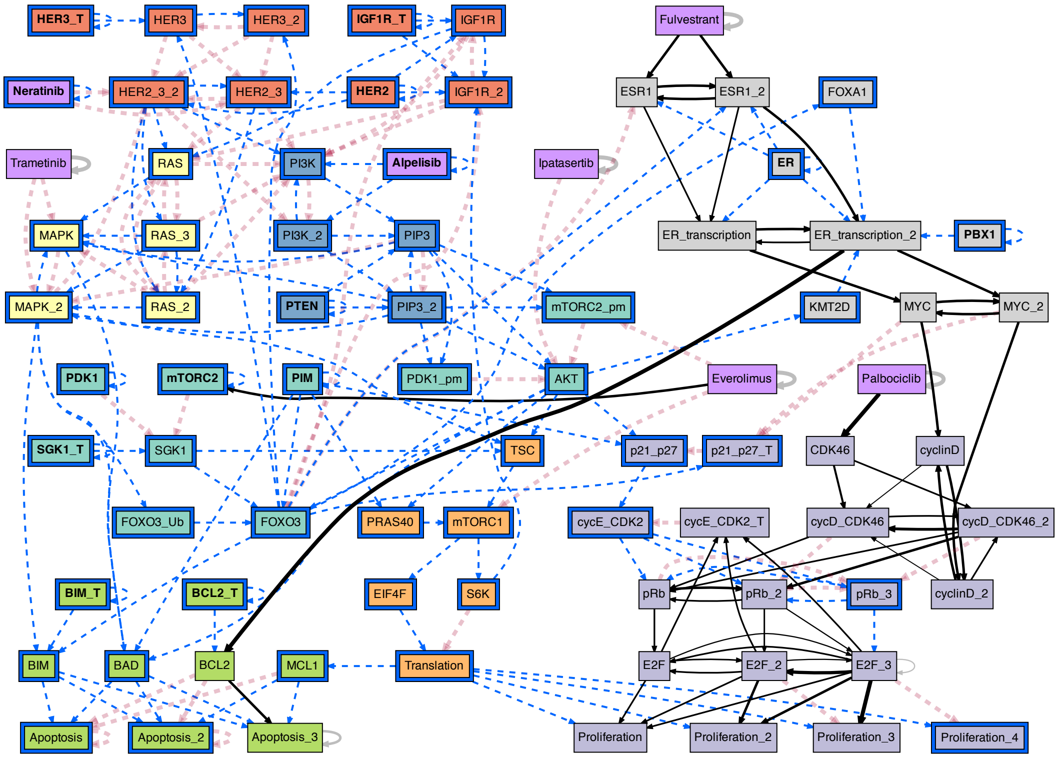


Fig. S20. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Neratinib*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Neratinib = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Neratinib$ does resolve any additional dynamics to what $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline already do (Fig. S16).

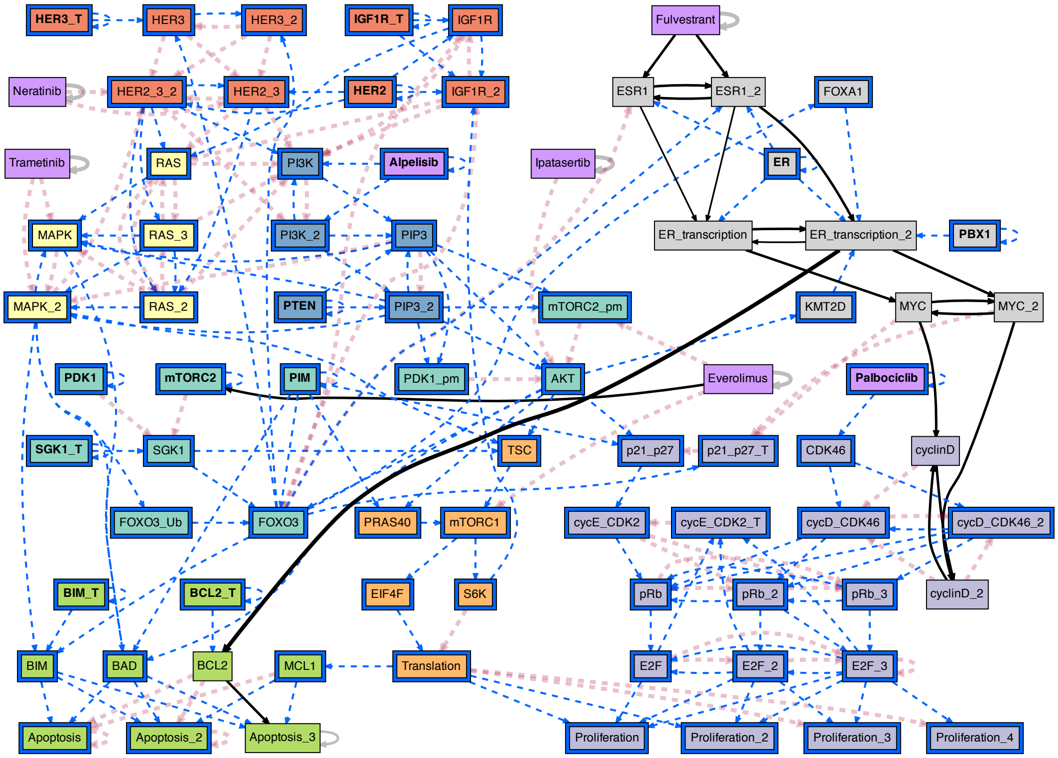


Fig. S21. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Palbociclib*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Palbociclib = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Palbociclib$ resolves the proliferation pathway in addition to the dynamics that $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline resolved on their own (Fig. S16. In particular, every proliferation node gets resolved, which means that $Palbociclib$ displays useful synergy with $Alpelisib$. However, unlike the case of $Fulvestrant$, $Palbociclib$ has no effect on apoptosis in this model for the $ER+/Her2-$ cancer cell state baseline.

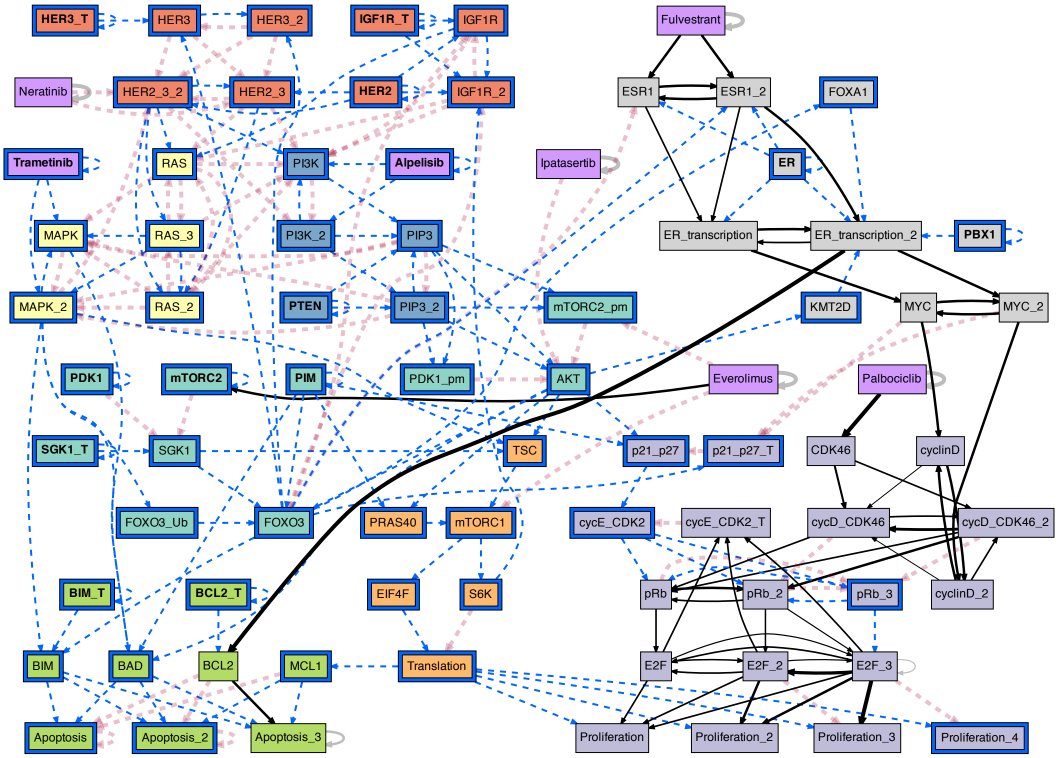


Fig. S22. The conditional effective graph of the ER+ breast cancer BN model, conditioned on the *ER+/Her2-* cancer cell state baseline + *Alpelisib=Trametinib*=ON. K is comprised of baseline nodes (see S15 caption) $+\{Alpelisib = \text{ON}, Trametinib = \text{ON}\}$. Variables in K (those initially pinned) are shown with a blue border and bold text; variables whose state becomes fixed (become constants), are shown with a blue border only. Edges that transmit a constant input state are denoted by a dashed blue color, while unresolved edges are denoted by black color with thickness proportional to their effectiveness, e_{ji}, with the fully redundant edges shown in dashed red. We can see that $Trametinib$ does resolve any additional dynamics to what $Alpelisib$ with the $ER+/Her2-$ cancer cell state baseline already do (Fig. S16).

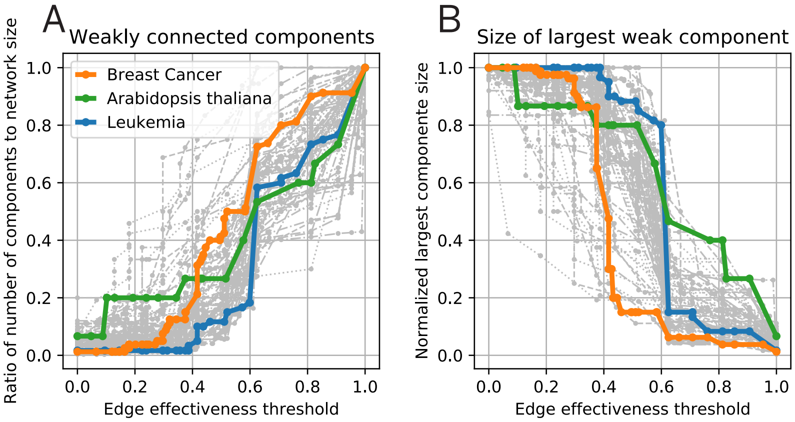


Fig. S23. Weakly connected components of threshold effective graphs reveal dynamical modules. **A** Ratio of the number of weakly connected components to network size. **B** Size of the largest weakly connected component, normalized by network size. In both graphs, the *ER+* breast cancer (orange), leukemia (blue), and *Arabidopsis thaliana* (blue) networks are highlighted.

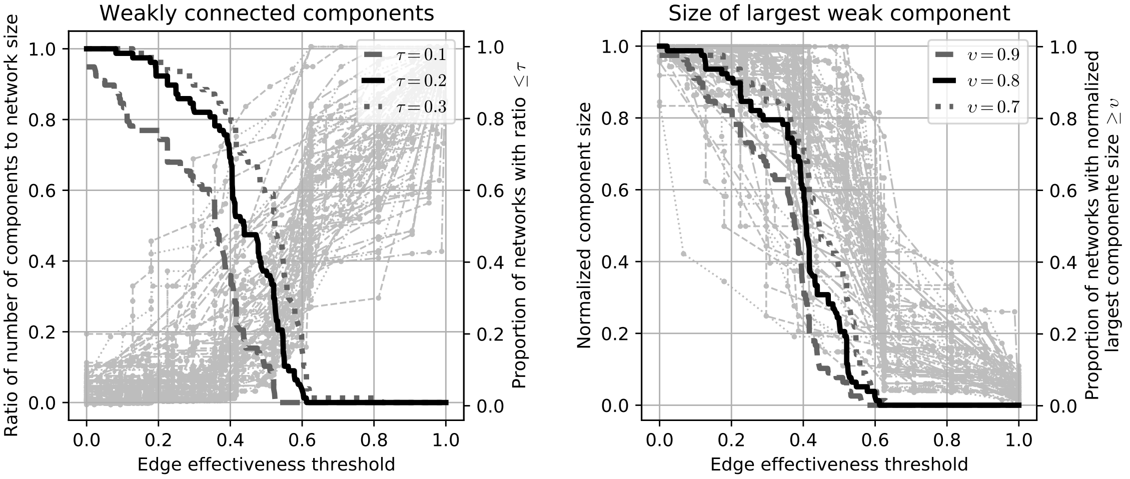


Fig. S24. Analysis of weakly connected components per edge effectiveness threshold for all Cell Collective models. Gray thin lines denote each network in dataset, thick lines denote overall statistics per legend. **Left.** Ratio of number of weakly connected components to network size (left vertical axis) for a given edge effectiveness threshold (horizontal axis); ratio is 1 when every node in network is a separate component, and very small when there is a single weakly connected component. Also shown are three statistics for the proportion of networks with ratio $\leq \tau$ (right vertical axis); e.g. for $\tau = 0.2$, black thick line denotes the proportion of networks whose ratio of number of weakly connected components to network size is smaller than 0.2 at a given edge effectiveness threshold. We can see, for instance, that for an edge effectiveness of 0.2, more than 90% of the networks have a small number of weakly connected components, specifically, less than $\tau = 20\%$ of the network size; for edge effectiveness larger than 0.4, on the other hand, most networks quickly break into many components, and for edge effectiveness larger than 0.6, no networks have a ratio of number of components to network size smaller than 20% (or even 30%, per dotted thick line) . **Right**. Size of largest weakly connected component relative to network size (left vertical axis) for a given edge effectiveness threshold (horizontal axis); ratio is 1 when there is a single weakly connected component, and very small when every node is its separate component. Also shown are three statistics for the proportion of networks with largest normalized component size $\geq \upsilon$ (right vertical axis); e.g. for $\upsilon = 0.8$, black thick line denotes the proportion of networks whose largest normalized component size is larger than 0.8 at a given edge effectiveness threshold. We can see, for instance, that for an edge effectiveness of 0.2, about 90% of the networks have a largest weakly connected component comprised of at least $\upsilon = 80\%$ of the network; for edge effectiveness larger than 0.4, on the other hand, most networks quickly lose a substantial largest weakly connected component, and for edge effectiveness larger than 0.6, no networks have a largest weakly connected component comprised of at least 80% of the network (or even 70%, per dotted thick line)