Supplementary Information for Advancing network resilience theories with symbolized reinforcement learning

Contents

Supplementary Notes

1 Quantifying resilience for representative networks

To provide an illustrative example on the ability of our framework to accurately estimate network resilience κ and keystone nodes V_c, we evaluate it across four representative complex networks. These networks encompass both cellular and neuronal dynamics and span both synthetic and real network topologies. For each network, starting from its original topology, we enumerate all possible induced topology by removing varying number of its nodes and assess their resilience, which yields a diagram of the resilient and non-resilient regions. We adopt topology (average degree) and dynamic (average b_i) as the two dimensions in this diagram, respectively (see Supplementary Figure 1). Consequently, when operating on a network by removing one node at a step, we jump from one point to another in the topology-dynamic-resilience diagram, and the process concludes upon reaching a non-resilient point. Moreover, to estimate the resilience κ, we need to discover the shortest path from the starting resilient point to the non-resilient region, minimizing the number of jumps needed. We utilize the discovered $d \cdot s$ formula to accomplish the transition from resilience to non-resilience, and compare it with existing physical metrics[1–3] and AI models[4,5]. As demonstrated in Supplementary Figure 1, the derived $d \cdot s$ theory indeed consumes fewer number of jumps to achieve the non-resilient region, discovering the shortest distance to losing the network's resilience, with a substantial 20%-50% reduction compared with all existing approaches (see Supplementary Table 1 for full results). Meanwhile, besides the original complete topology, we also evaluate the performance starting from the residual topologies after a few attacks by the resilience centrality (RC) approach[1]. As evident in Supplementary Figure 1, $d \cdot s$ takes distinct routes from RC and requires fewer steps to reach the termination point, consistently delivering more precise estimations of κ.

In Supplementary Figure 1, the paths resulted from existing approaches–diverging from and much longer than the path by the $d \cdot s$ formula–indicate that existing theoretical and AI approaches identify the incorrect critical nodes and highlight the effectiveness of the $d \cdot s$ formula. Indeed, we demonstrate the selected critical nodes in Supplementary Figure 2 of the four representative networks by different approaches, which display substantial differences. Particularly, both RC and FINDER remove much more nodes and the remaining network almost disintegrates. For instance, in Supplementary Figure 2a and 2d, the residual topologies by RC and FINDER only contain fewer than 5 nodes, while our method compromises the two networks' resilience with their structure maintained, keeping 11 and 12 nodes, respectively.

In Supplementary Figure 2b and 2d, we run different system dynamics (cellular and neuronal) on the same network topology. Indeed, system dynamics exert a crucial impact on network resilience, leading to distinct keystone nodes V_c for various $[\mathbf{F}, \mathbf{G}]$, even within the same topology $\mathbf{A}$. While the keystone nodes V_c identified by current approaches[1–5], all within the context of $\mathcal{Q}(i; \mathbf{A})$ focusing on topology while ignoring system dynamics, remain consistent across various $[\mathbf{F}, \mathbf{G}]$ as long as they operate atop the same $\mathbf{A}$. As both RC and FINDER ignore the crucial influence of system dynamics $[\mathbf{F}, \mathbf{G}]$, their corresponding node importance functions $\mathcal{Q}$ lie under the context of $\mathcal{Q}(\mathbf{A})$, thus the selected nodes by these approaches are the same, as illustrated in Supplementary Figure 2b and 2d. On the contrary, our method takes the dynamics $[\mathbf{F}, \mathbf{G}]$ into consideration, resulting in a comprehensive node importance function $\mathcal{Q}(\mathbf{A}, [\mathbf{F}, \mathbf{G}])$, which can identify different keystone nodes according to the system dynamics, with the number of removed nodes much fewer than RC and FINDER. The mis-identification of V_c and wrong estimation of κ by existing approaches highlight a significant gap: the current theories of network resilience are designed to treat topological importance only, exposing severe limits to our ability to achieve a comprehensive understanding of network resilience with complicated network dynamics involved.

2 Full results on real-world large networks

Our model is trained with synthetic networks containing fewer than 100 nodes, while the derived theory holds universal effectiveness from the experimental environment to practical systems. To prove this, we assess the performance of our model on real-world networks, particularly large-scale networks, in comparison with current baselines. Specifically, we utilize three real-world large networks released by ref[6], including two gene regulatory networks (Human and Yeast) and one neuronal network (Brain). The three networks contain 3125, 1647, and 989 nodes, respectively. Besides the original networks, we also extract sub-graphs from them via community detection, generating multiple test cases with varying sizes, ranging from a dozen to several hundred nodes. We leverage both Fluid Communities algorithms[7] and Louvain Community Detection Algorithm[8] to extract sub-graphs of different sizes by varying the number of clusters and setting different resolutions.

As illustrated in Supplementary Table 2, our approach significantly enhances the precision of quantifying network resilience κ, achieving an average reduction of over 42.2% compared to the best baseline. The maximum improvement even surpasses 77.8%, revealing the large errors of existing approaches when applied to practical systems. Notably, as the network size increases, the problem complexity grows exponentially, making it exceedingly challenging to identify critical nodes in large networks. For the three original complete networks, our method achieves an average improvement of over 49.1% and a maximum improvement of over 74.4%. The consistent improvements of our approach on real-world networks with varying sizes validate the scalability of the universally effective $d \cdot s$ metric, implying its promising applicability in real systems.

3 Full results on synthetic networks

3.1 Synthetic networks

We evaluate our method across a diverse array of synthetic networks, including ER networks, BA networks, RP networks, and SW networks. All these synthetic networks are generated using networkx[9], with their sizes ranging from 80 to 200. For each case, we generate 10 networks by setting different random seeds. For the RP networks, we adopt two randomly partitioned communities of equal sizes $\frac{N}{2}$, and set the probability of edges within and between communities as $\frac{2\langle d\rangle}{N}$ and $\frac{\langle d\rangle}{5N}$, respectively. For the SW networks, we set the probability of rewiring each edge as 0.4. For both ER and BA networks, we set the network generation parameters according to the predefined average degree ($\langle d\rangle$).

3.2 System dynamics

We investigate cellular dynamics and neuronal dynamics which are characterized by the following coupled equations,

$$\text{cellular:} \quad \frac{\mathrm{d}x_i}{\mathrm{d}t} = -b_i x_i^f + \sum_{j=1}^{N} A_{ij} \frac{x_j^h}{1+x_j^h}, \tag{1}$$

$$\text{neuronal:} \quad \frac{\mathrm{d}x_i}{\mathrm{d}t} = -b_i x_i + \sum_{j=1}^{N} A_{ij} \frac{1}{1+e^{\mu-\delta x_j}}, \tag{2}$$

For cellular dynamics, we set Hill coefficient $h = 2$ for the cooperation level and $f = 1$ for degradation accordingly[10]. For neuronal dynamics, we set $\mu = 3.5$ and $\delta = 2.0$ accordingly[6]. We achieve the dynamical heterogeneity via imposing a power-law to the self decay rate b_i,

$$f(x,a) = a \cdot x^{a-1}, 0 \le x \le 1 \tag{3}$$

$$b_i = b + s \cdot x_i, \tag{4}$$

where b and s are the base value and scaling factor, and a controls the level of heterogeneity. Specifically, $a = 1$ indicates the most heterogeneous case where b_i is distributed uniformly in $[b, b+s]$, while setting a to a large value such as 10 implies dynamical homogeneity where almost all the nodes share the same dynamical parameter b_i as $b+s$.

3.3 Results on cellular dynamics

Resilience demonstrates universal patterns across various network dynamics and topologies[10], and we anticipate that the derived $d \cdot s$ metric inherits similar universality in different scenarios. As introduced previously, the metric is derived from synthetic SF networks of 80 nodes (using cellular dynamics), while real-world systems often exhibit diverse topological connection characteristics, such as the community structures, or the small-world effect[11,12]. In the paper, we show the universal effectiveness of the derived $d \cdot s$ formula across a wide range of networks (ER, BA, RP, and SW networks), comparing with both theoretical and AI approaches, including resilience centrality[1] and FINDER[4]. Here we provide the full results with more baselines. Specifically, we include both physical metrics[1–3] and AI models[4,5]. As illustrated in Supplementary Figure 4, our approach demonstrates substantial improvements over all existing baselines, achieving an average reduction of over 43.25% in the estimated network resilience κ. In particular, black-box AI models, including FINDER[4] and GDM[5], demonstrate no advantages over traditional physical metrics like RC[1] and degree centrality (DC), since both categories of approaches fail to capture the non-linear and heterogeneous dynamics, a crucial factor influencing the network resilience. The derived $d \cdot s$ metric takes both topology and dynamics into consideration, and the consistent improvements observed across four distinct types of network topologies affirm the universal ability of our approach in precisely estimating the resilience κ.

3.4 Results on neuronal dynamics

We then study the universality of $d \cdot s$ by varying the network dynamics. In Supplementary Figure 5 we demonstrate the estimation performance for neuronal dynamics on ER, BA, RP and SW networks. Though derived from the gene regulatory dynamics, the ability of our approach to precisely estimate κ is still valid for the distinct neuronal dynamics, and it outperforms all baselines, with the estimated κ reduced substantially by over 56.2% on average. The experiments across different network topologies and system dynamics highlight the superior universality across various scenarios of the derived $d \cdot s$ formula.

4 Effectiveness of self-inductive symbolized reinforcement learning

4.1 Search

We utilize 30 scale-free (SF) networks of 80 nodes to train the RL agent. Specifically, we first generate the degree sequence from a powerlaw ($\gamma = 2.0$), then scale the sequence according to a predefined average degree value ($\langle d\rangle = 6$), and finally round

them up to integers. With the degree sequence, we construct a *pseudograph* by randomly assigning edges to match the given degree sequence, which possibly contains parallel edges and self loops. Finally, we remove these parallel edges and self loops, resulting in the undirected training graphs.

We train the RL agent to attack the generated synthetic networks with the widely adopted Python library Stable Baselines3[13], using gene regulatory dynamics. To illustrate the effectiveness of our RL agent, we include three baseline methods for reference, which are degree centrality (DC), resilience centrality (RC) and FINDER. The average estimated reslience (the number of attacked nodes) of the three baselines are 10.30, 9.77, 9.80, respectively. In Supplementary Figure 6a, we show the convergence of the mean estimated resilience of the RL model during the training process. Specifically, the RL agent successfully outperforms baselines methods after only 25,000 training steps, and continue to improve its estimation performance as the solution space is more sufficiently explored. Eventually, the RL agent achieves an average estimated resilience of 7.63, significantly less than the results of baselines, with a remarkable improvements of 21.9%.

4.2 Distill

In order to understand how the RL agent attains the estimation, we analyze the importance of different node features using the 30 training networks for the RL model with GNNExplainer[14], calculating the contribution of each node feature to the model prediction. After obtaining the feature importance scores, we normalize them to the range $[0,1]$ by subtracting the minimum importance score and dividing over the gap between the maximum and minimum importance score, such that the most and the least important features have the score of 1 and 0, respectively. As demonstrated in Supplementary Figure 6b, the features that have significant influence on the model output are `degree`, `neighbor degree`, `resilience centrality`, `neighbor state`, `state`. Since the feature `resilience centrality` can be calculated by combining `degree` and `neighbor state`, we omit it to reduce redundancy. Therefore, the identified important features are thus `degree`, `neighbor degree`, `neighbor state`, `state`. It is worth noting that, though `degree`-related features are commonly adopted by existing methods, `state`-related features are often ignored which contain valuable information about the network dynamics, playing important roles for network resilience. By employing XAI techniques, we distill the dominant ingredients that drive the decision process of the intricate RL model.

4.3 Discover

To fully reveal the underlying rules of the RL model, we connect important features to the network resilience with a mathematical formula using symbolic regression (SR). Using the 30 synthetic networks for training the RL model, we construct an SR dataset, containing the important node features identified by XAI and labels indicating whether each node is selected by the RL model. With this dataset, we employ the efficient PySR library[15] to explain the actions of the RL model, and finally achieve tangible equations that describe the contribution of each node to the overall network resilience. Table 3 illustrates the discovered equations from SR, where we order these formulas from the least complex (highest error) to the most complex (lowest error). The equations generated by SR may appear complex in their raw form, hence we involve human experts to refine these formulas to achieve the final formula. Specifically, certain terms are eliminated according to the dimensions. Meanwhile, constant values are omitted as they do not affect the relative order of different nodes. Finally, as illustrated in Supplementary Figure 6c, striking a balance between accuracy and simplicity, we obtain a novel formula denoted as $d \cdot s$, the product of `degree` and `state`, which resides at the core of the black-box RL agent, offering valuable insights into the individual contributions of each node to the network's resilience.

5 Improving and reproducing classical physical formulas

Though designed for quantifying network resilience κ and keystone nodes V_c, our proposed framework is not limited to this specific task. Instead, the self-inductive symbolized reinforcement learning framework serves as a versatile tool for knowledge discovery in complex networks. With proper definition and adaption, our framework can solve problems that are analogous to equations (2-5).

5.1 Improving over resilience centrality

We investigate network resilience in dynamical homogeneous scenarios, featuring $F_i = F, \forall i$ and $G_{ij} = G, \forall ij$. With dynamical homogeneity, the resilience function $\mathcal{R}$ can be reduced to a 1-D equation[10], which leads to a theoretical solution of $\mathcal{Q}$ as resilience centrality[1]. To demonstrate the universal ability of our framework in scientific discovery, we run the self-inductive symbolized RL framework in dynamical homogeneous scenarios. We include six node features, namely `degree`, `maximum edge weight`, `neighbor degree`, and `dynamics parameters 1/2/3`. In Supplementary Figure 7a we illustrate the normalized node importance scores of the policy network, where the two dominant features are `degree` and `neighbor degree`. We utilize these two features as primitives for SR in the discover process to predict the identified keystone nodes V_c by the RL agent, leading to the candidate formulas in Supplementary Table 4. In the last three regressed

formulas, one critical term emerges, in the form of $2\bar{d}+d\cdot(\bar{d}-C)$, improving over the resilience centrality index by one term denoted as $2\bar{d}+d\cdot(d-C)$, which is originally obtained through heavy theoretical derivations and analysis.

5.2 Reproducing universal resilience function

Besides studying the policy network, we also delve into the value network which measures the resilience conditions of the current network. Indeed, in dynamical homogeneous scenarios, the resilience function $\mathcal{R}$ can be expressed as a concise empirical expression called the β equation[10]. We are able to faithfully re-discover this formula using the self-inductive symbolized RL framework. Specifically, we distill the value network of the RL agent, and in Supplementary Figure 7b we illustrate the feature importance scores. Again, the two features `degree` and `neighbor degree` contribute significantly to the prediction of the value network. We then employ the similar SR process to decipher the value network despite we replace the target of SR from keystone nodes V_c to the output of the value network. Surprisingly, our self-inductive framework successfully reproduces the β equation, as demonstrated in Supplementary Table 5 where two candidate equations exhibit the critical term, $\frac{\langle d^2\rangle}{\langle d\rangle}$, which is just the same as the expression of β in its original proposal[10].

Both resilience centrality and β were achieved manually by exhaustive theoretical efforts, while they are now successfully reproduced or even improved in a computational rather than theoretical way. More importantly, in complicated problems such as the quantification of network resilience κ and keystone nodes V_c with non-linear and heterogeneous dynamics involved in this work, theoretical treatments may become inapplicable due to the non-analytic nature of the problem, thus an AI framework being able to produce tangible and human-understandable formulas can greatly benefit and accelerate scientific discovery. Further efforts can made to extend the self-inductive framework to discover novel mechanisms besides network resilience, and we believe that the proposed framework may serve as a model for future AI-enabled scientific discovery.

6 Advantages of mathematical formulas over black-box AI models

A black-box AI model alone may provide accurate predictions, albeit it has limited explainability compared to a tangible formula, posing a challenge in comprehending the internal mechanisms behind its predictions. More importantly, beyond the aspect of explainability, a physical insightful formula exhibits greater robustness across different scenarios, underscoring its ability to extrapolate insights to previously unseen data. Specifically, the powerful expressive capabilities of neural networks enable AI models to fit intricate functions within the dataset, including the inherent noise. Such noise often signifies unstable relationships, constraining the generalization ability of AI models and impeding their predictive performance on unfamiliar data. In our experiments, we train the RL model utilizing 30 SF networks. It is likely that the RL model captures patterns specific to these 30 cases, which may not be truly representative of general scenarios, leading to failures in estimating resilience of dissimilar networks.

In order to assess the generalization ability, we evaluate the performance of resilience estimation on training networks present in the training dataset and test networks that are not seen during the training process. We compare the black-box RL model obtained after the search process and the derived $d\cdot s$ formula following the complete self-inductive framework. We demonstrate in Supplementary Figure 8a the performance of RL and $d\cdot s$ on training and test networks, with the relative difference between the estimated network resilience κ by the two approaches. As expected, with strong ability in fitting training data, the RL model achieves slightly better performance than the $d\cdot s$ metric on training networks. However, in the crucial task of generalizing to unseen test networks, $d\cdot s$ demonstrates substantial advantages in estimation precision. In particular, as the network size increases, the gap between $d\cdot s$ and the RL model becomes larger, with the most significant improvement of κ by 10 nodes. The results verify the superior generalization ability of the $d\cdot s$ metric in contrast to the RL model. Moreover, the discrepancy between RL and $d\cdot s$ affirms the necessity of obtaining an explainable and tangible formula, which is not accomplished by existing *AI for science* approaches that often culminate in black-box models. On the contrary, our proposed self-inductive framework, distinct from prior approaches, delves deeper into the success of AI models, unraveling the underlying rules governing the RL model's decision-making process.

Besides generalizing towards unseen data and delivering valuable insights, a mathematical formula also displays super-fast inference speed in comparison to the complex computations of multi-layer neural networks in the RL model. In Supplementary Figure 8b we illustrate the average inference time over 10 networks of the RL model Θ and the mathematical formula θ, across varying network sizes. As expected, the mathematical formula exhibits substantially shorter inference time than the RL model, with an average inference time reduction of over 34.5%. Notably, for larger networks with 160-200 nodes, the time reduction is even more significant, exceeding 50.1%.

7 Network protection

Network resilience can be effectively enhanced by safeguarding keystone nodes identified by the derived $d \cdot s$ formula. Here we provide examples of this strategic network protection in Supplementary Figure 9 and 10 for cellular and neuronal networks, respectively. For cellular networks, safeguarding 3 nodes with the largest $d \cdot s$ values boosts the network resilience κ for over 2.24 times in average. Similarly for neuronal networks, safeguarding the top 3 nodes according to the $d \cdot s$ formula can increase network resilience κ by about 2.23 times. Surprisingly in Supplementary Figure 9b-c, Supplementary Figure 10a and Supplementary Figure 10c, we observe that the safeguarded nodes eventually detach from the network such that the network loses its resilience. As these protected nodes can not be directly removed, their disconnections actually result from the removal of all their neighbors. In other words, these keystone nodes with high $d \cdot s$ play dominant roles in maintaining the system resilience, thus they must be removed to compromise a network's resilience, either by directly removing them or by removing their neighbors to get rid of these nodes to block their influence. The latter one usually takes much more efforts due to the inter-connectivity of the network structure (these keystone nodes tend to connect multiple nodes), thus safeguarding nodes according to $d \cdot s$ serves as an efficient strategy for enhancing network resilience.

Supplementary Tables

Supplementary Table. 1. Full results of the estimated resilience κ on 4 typical networks. F means failing to compromise the network's resilience.

Dynamics	Network	DC	RC	GND	EI	GDM	FINDER	Ours	impr%
Cellular	Real	6	6	F	F	6	5	**4**	**20.0%**
	Synthetic	7	5	6	9	11	5	**4**	**20.0%**
Neuronal	Real	6	6	13	10	9	9	**3**	**50.0%**
	Synthetic	7	7	5	9	6	6	**3**	**40.0%**

Supplementary Table. 2. Full results of the estimated network resilience κ on real-world large-scale networks. S means sub-graphs extracted via community detection, and G means the original complete graph.

Dynamic	Network	ID	N	DC	RC	GND	EI	GDM	FINDER	Ours	impr%
Cellular	Human	S-1	298	7	7	134	154	9	8	**6**	**14.3%**
		S-2	353	23	20	213	174	74	24	**16**	**20.0%**
		S-3	355	9	9	174	117	34	9	**4**	**55.6%**
		S-4	358	6	6	128	61	45	6	**5**	**16.7%**
		S-5	441	16	16	275	176	68	16	**14**	**12.5%**
		S-6	529	32	33	242	269	93	33	**12**	**62.5%**
		S-7	549	22	23	259	315	82	22	**11**	**50.0%**
		S-8	587	31	32	355	293	67	32	**15**	**51.6%**
		S-9	683	25	25	392	334	142	25	**19**	**24.0%**
		S-10	932	54	53	489	417	78	48	**19**	**60.4%**
		S-11	978	64	64	557	470	195	67	**32**	**50.0%**
		S-12	1136	78	75	755	644	240	83	**36**	**52.0%**
		G	3125	272	256	2026	1924	507	263	**125**	**51.2%**
	Yeast	S-1	296	3	4	52	61	4	4	**2**	**33.3%**
		S-2	340	10	10	99	126	29	9	**2**	**77.8%**
		S-3	491	17	16	213	192	24	17	**11**	**31.3%**
		S-4	507	25	16	247	178	39	23	**4**	**75.0%**
		S-5	701	30	38	281	294	51	31	**8**	**73.3%**
		S-6	773	34	29	247	321	58	34	**14**	**51.7%**
		S-7	931	50	43	403	433	80	52	**16**	**62.8%**
		G	1647	121	118	808	726	193	117	**30**	**74.4%**
Neuronal	Brain	S-1	286	103	105	180	258	203	205	**84**	**18.4%**
		S-2	313	119	119	215	206	212	226	**96**	**19.3%**
		S-3	359	158	155	236	267	251	210	**122**	**21.3%**
		S-4	676	278	271	351	541	490	446	**222**	**18.1%**
		G	989	428	430	490	889	745	543	**335**	**21.7%**

Supplementary Table. 3. Output equations of SR, ordered by complexity from low to high (or by loss from high to low). Equations are refined and simplified manually according to dimensions, and constant terms are discarded as they do not affect the relative order of different nodes.

Loss	Regressed equation	Expanded equation	Simplified metric
0.2114	-434.27	-434.27	1
0.1388	$(-5.76+d)$	$d-5.76$	d
0.1177	$(-20.37+(d\cdot s))$	$d\cdot s-20.37$	$d\cdot s$
0.1159	$(-27.40+((s\cdot d)+d))$	$d\cdot s+d-27.40$	$d\cdot s$
0.1103	$(-9.81+((-0.39\cdot\bar{d})+(s\cdot d)))$	$d\cdot s-0.39\bar{d}-9.81$	$d\cdot s$
0.1100	$(-9.81+((-0.38\cdot\bar{d})+(s\cdot(-0.12+d))))$	$d\cdot s-0.12s-0.38\bar{d}-9.81$	$d\cdot s$
0.1093	$((-9.81+((-0.66+(-0.38\cdot\bar{d}))+(s\cdot d)))\cdot\bar{d})$	$d\cdot s\cdot\bar{d}-0.38\bar{d}^2-10.47\bar{d}$	$d\cdot s\cdot\bar{d}$
0.1090	$(((-9.81+((-0.66+(-0.38\cdot\bar{d}))+(s\cdot d)))\cdot\bar{d})+1.11)$	$d\cdot s\cdot\bar{d}-0.38\bar{d}^2-10.47\bar{d}+1.11$	$d\cdot s\cdot\bar{d}$
0.1087	$((((s\cdot d)+(-1.42+\bar{d}))+(-1.26\cdot(\bar{d}+(s+d)))\cdot\bar{d})$	$d\cdot s\cdot\bar{d}-0.26\bar{d}^2-1.26\bar{d}\cdot s-1.26d\cdot\bar{d}-1.42\bar{d}$	$d\cdot s\cdot\bar{d}$
0.1086	$(((((s\cdot d)+(-1.42+\bar{d}))+(-1.26\cdot(\bar{d}+(s+d)))\cdot\bar{d})\cdot\bar{d})$	$d\cdot s\cdot\bar{d}^2-0.26\bar{d}^3-1.26\bar{d}^2\cdot s-1.26d\cdot\bar{d}^2-1.42\bar{d}^2$	$d\cdot s\cdot\bar{d}^2$

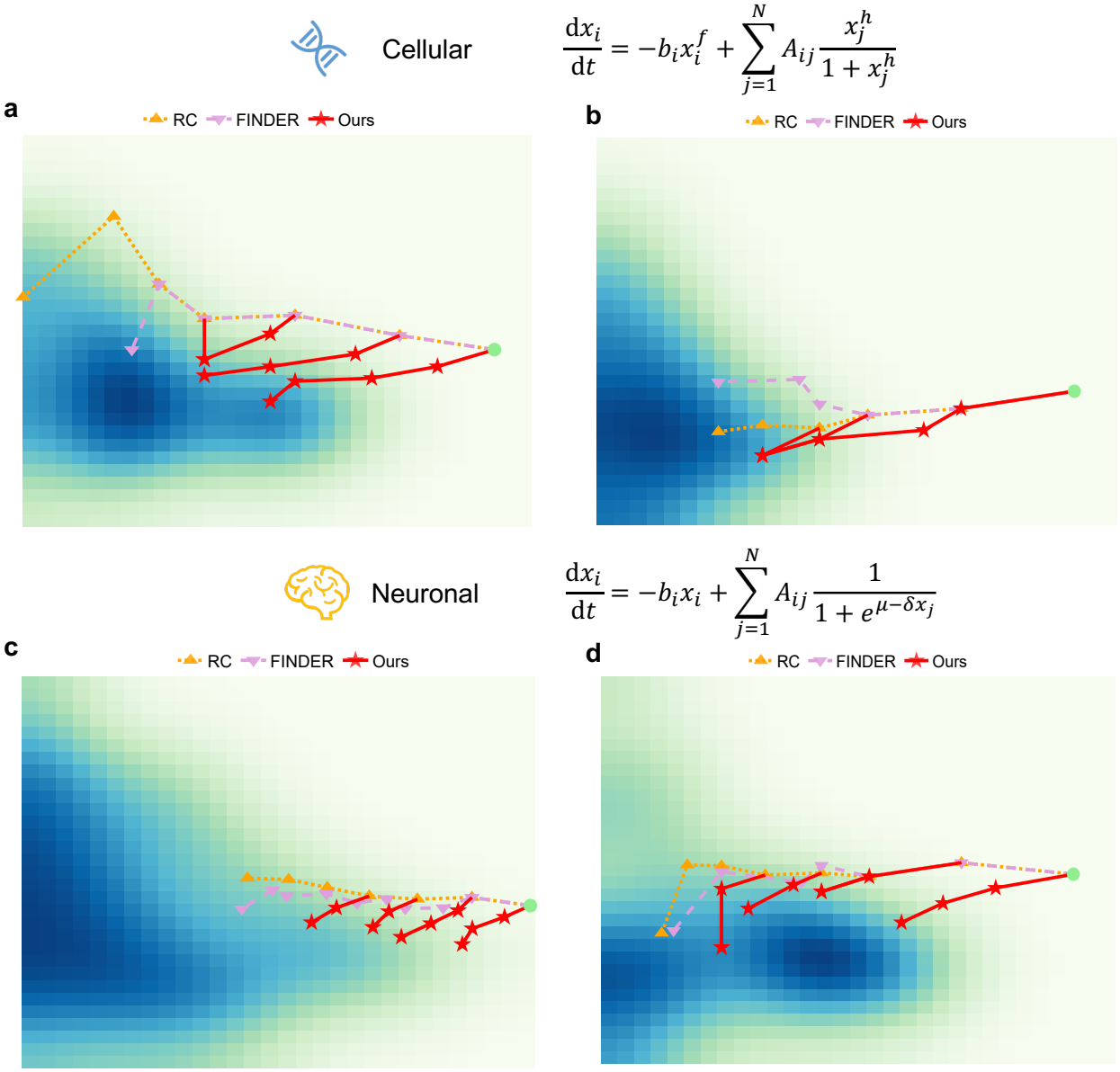
Supplementary Table. 4. Output equations of SR on the policy network that is trained under dynamical homogeneous conditions, ordered by complexity from low to high (or by loss from high to low). Equations are refined and simplified manually according to dimensions, and constant terms are discarded as they do not affect the relative order of different nodes. Critical terms of the equations are extracted to improve over resilience centrality of the form $2\bar{d}+d\cdot(d-C)$ where C is a constant value.

Loss	Regressed equation	Critical term
0.2184	136.8	1
0.1980	$(\bar{d}\cdot\bar{d})$	$\bar{d}^2$
0.1096	$((-2.0758+\bar{d})\cdot 53.743)$	$\bar{d}$
0.1084	$(((-1.9913+\bar{d})\cdot 92.329)\cdot d)$	$d\cdot\bar{d}$
0.0904	$((\bar{d}+((-4.2525+\bar{d})\cdot d))\cdot 136.8)$	$d\cdot\bar{d}$
0.0868	$(((\bar{d}+((-4.2525+\bar{d})\cdot d))\cdot 133.13)\cdot 3.4307)$	$d\cdot\bar{d}$
0.0863	$(((1.8274+((-4.2525+(\bar{d}+0.41679))\cdot d))\cdot\bar{d})\cdot 133.13)$	$d\cdot\bar{d}^2$
0.0862	$((((\bar{d}+((-4.2525+(-0.79803+\bar{d}))\cdot d))+\bar{d})\cdot 133.13)\cdot d)$	$2\bar{d}+d\cdot(\bar{d}-C)$
0.0862	$(1.4475+((((\bar{d}+((-4.2525+(-0.79803+\bar{d}))\cdot d))+\bar{d})\cdot 133.13)\cdot d))$	$2\bar{d}+d\cdot(\bar{d}-C)$
0.0861	$(((((((\bar{d}+((-6.7034+(-0.7394+\bar{d})))\cdot d))+\bar{d})\cdot 92.372)+168.51)\cdot d)+43.23)$	$2\bar{d}+d\cdot(\bar{d}-C)$

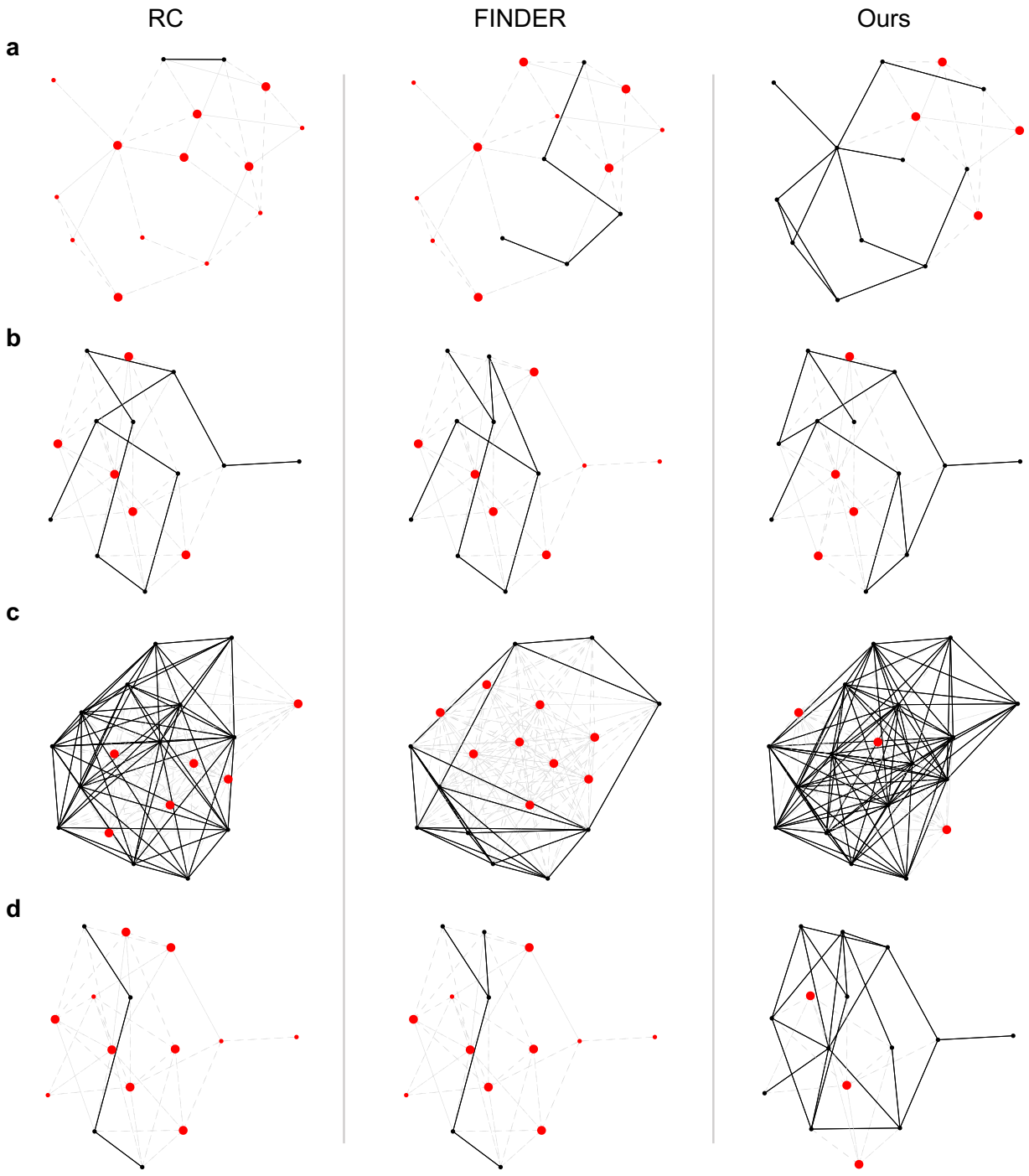
Supplementary Table. 5. Output equations of SR on the value network that is trained under dynamical homogeneous conditions, ordered by complexity from low to high (or by loss from high to low). Equations are refined and simplified manually according to dimensions, and constant terms are discarded as they do not affect the relative order of different nodes. Critical terms of the equations are extracted to reproduce β equation of the form $\frac{\langle d^2\rangle}{\langle d\rangle}$.

Loss	Regressed equation	Critical term
1.0500	$\langle d\rangle$	$\langle d\rangle$
0.0577	$(\langle d^2\rangle\cdot 0.14479)$	$\langle d^2\rangle$
0.0573	$(-0.032747+(0.14654\cdot\langle d^2\rangle))$	$\langle d^2\rangle$
0.0355	$((-1.6515+\langle d\rangle)+(0.081034\cdot\langle d^2\rangle))$	$\langle d\rangle$
0.0120	$((-2.1623+\langle d\rangle)+(0.33149\cdot(\frac{\langle d^2\rangle}{\langle d\rangle})))$	$\frac{\langle d^2\rangle}{\langle d\rangle}$
0.0120	$((-2.1827+\langle d\rangle)+(0.33151\cdot((\frac{\langle d^2\rangle}{\langle d\rangle})+0.061213)))$	$\frac{\langle d^2\rangle}{\langle d\rangle}$
0.0065	$((-0.60722+(\langle d^2\rangle\cdot(-0.0029165\cdot\langle d^2\rangle)))+(0.12484\cdot(\langle d^2\rangle+\langle d^2\rangle)))$	$\langle d^2\rangle$
0.0048	$((-0.75985+(\langle d^2\rangle\cdot(-0.0024985\cdot\langle d^2\rangle)))+(0.22046\cdot((-0.55267+\langle d\rangle)+\langle d^2\rangle)))$	$\langle d^2\rangle$
0.0048	$((-0.73231+((-0.98001+\langle d^2\rangle)\cdot(-0.0025045\cdot(-0.36081+\langle d^2\rangle))))+(0.2175\cdot(-0.66627+(\langle d^2\rangle+\langle d\rangle))))$	$\langle d^2\rangle$

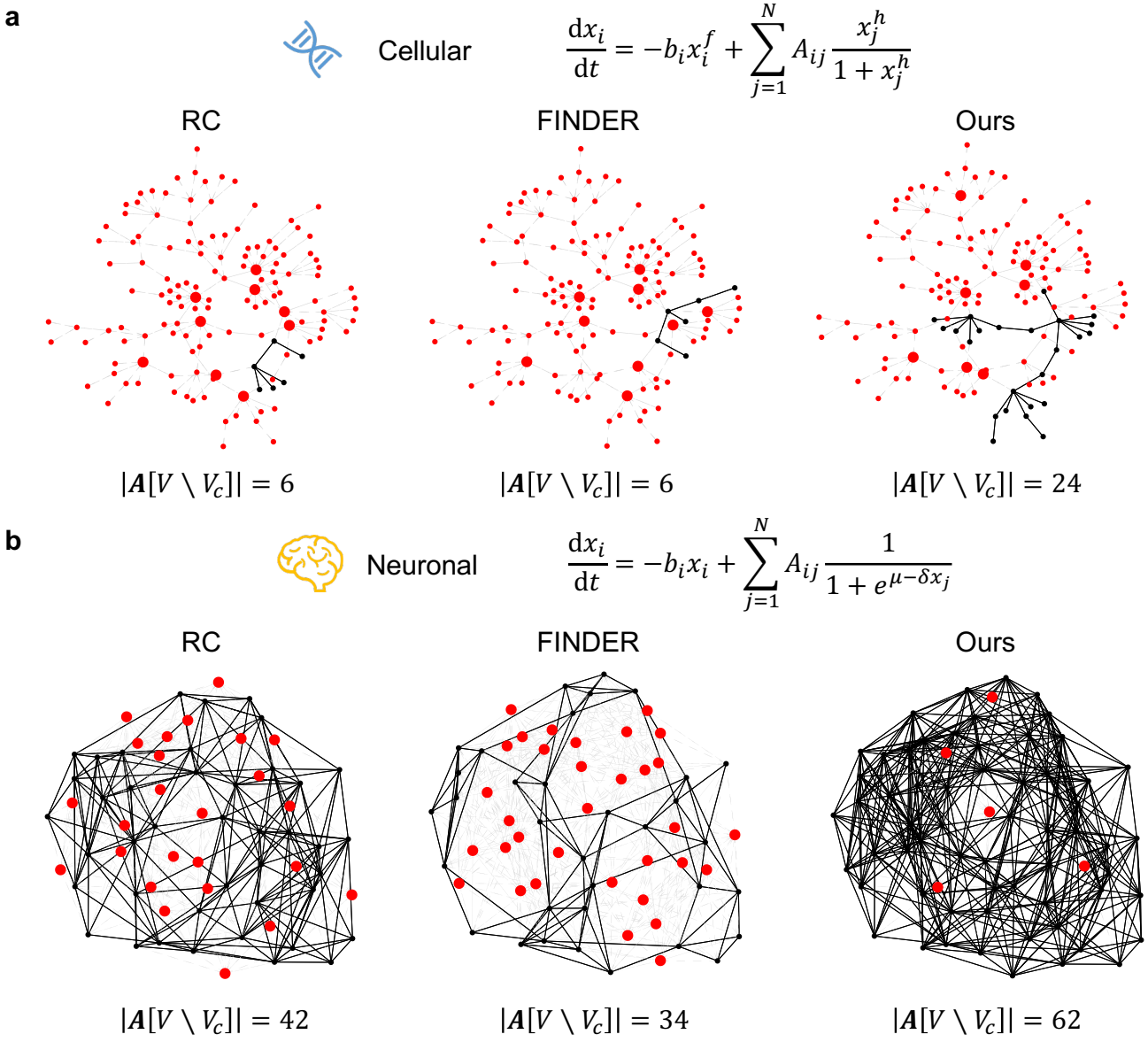
Supplementary Figures



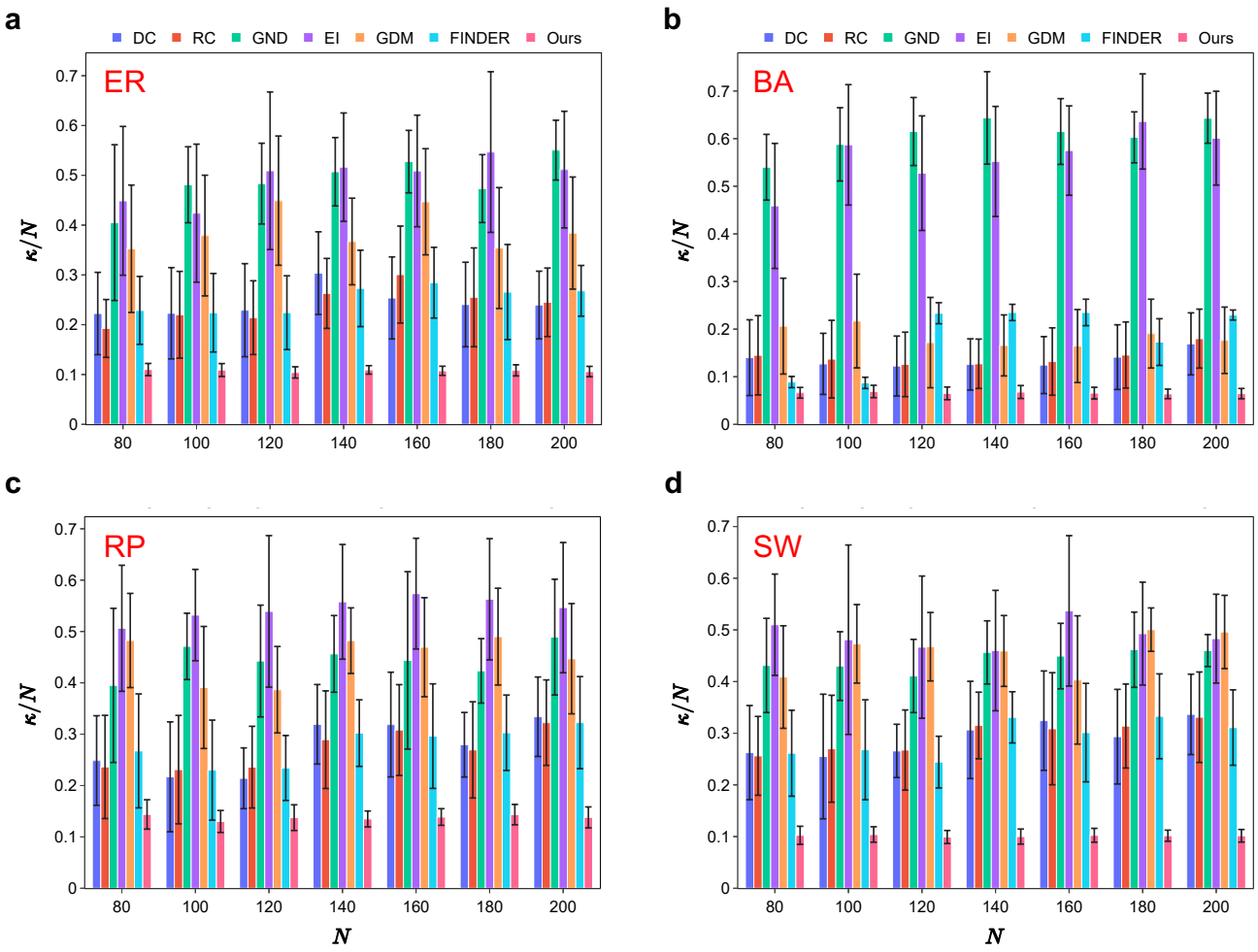
Supplementary Figure 1. Visualization of the node removal process by different methods. The green circle represents the original topology from which all methods start to remove one single node at a step, indicated by different symbols. The x-axis means the average degree and the y-axis means the average b_i value. The color of each cell represents the ratio of resilient networks. The experiments are conducted on the following four typical networks: **a.** a real cellular network. **b.** a synthetic cellular network. **c.** a real neuronal network. **d.** a synthetic neuronal network.



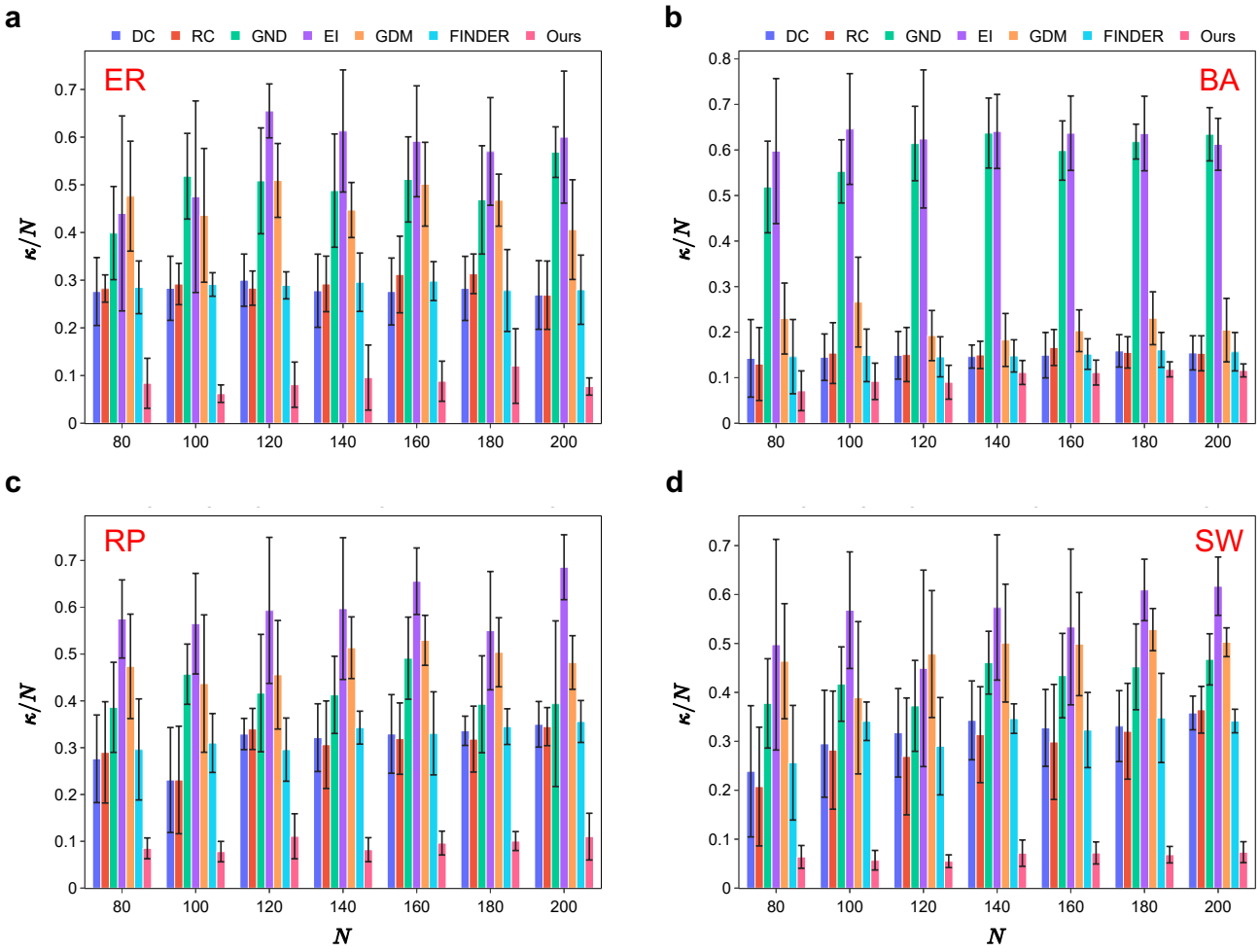
Supplementary Figure 2. Visualization of the selected nodes V_c and the remaining topology $\mathbf{A}[V \setminus V_c]$ by different approaches. Large red nodes are the selected nodes for removal, and small red nodes are the omitted nodes that are not in the greatest connected component (GCC) after node removal. Black nodes and edges are the remaining topologies. The experiments are conducted on the following four typical networks: **a.** a real cellular network. **b.** a synthetic cellular network. **c.** a real neuronal network. **d.** a synthetic neuronal network.



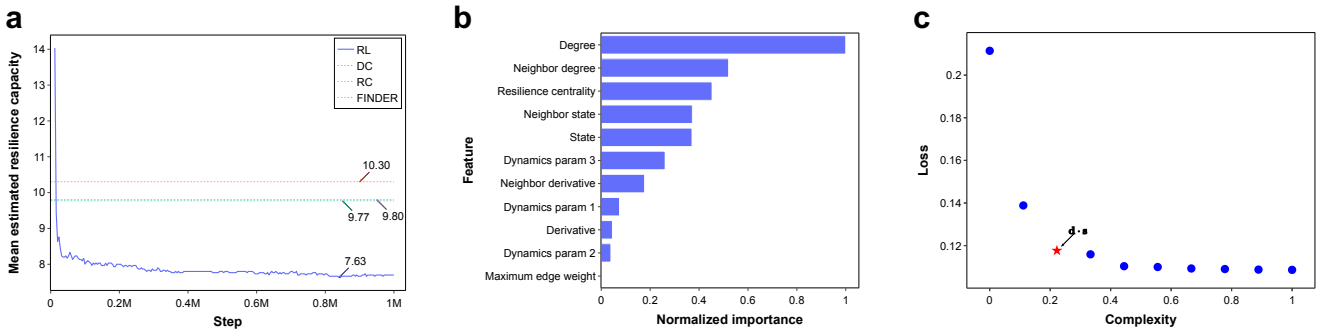
Supplementary Figure 3. Visualization of the selected nodes V_c and the remaining topology $\mathbf{A}[V \setminus V_c]$ by different approaches. Large red nodes are the selected nodes for removal, and small red nodes are the omitted nodes that are not in the greatest connected component (GCC) after node removal. Black nodes and edges are the remaining topologies. The experiments are conducted across two complex networks: **a.** a real cellular network. **b.** a real neuronal network.



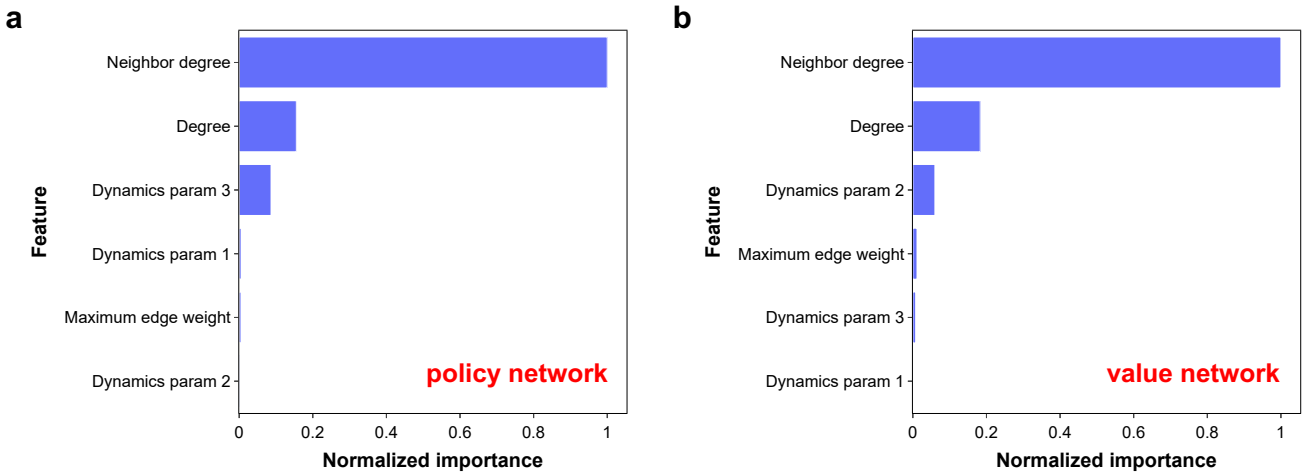
Supplementary Figure 4. The average quantified network resilience across 10 cellular networks of different methods for **a.** ER networks **b.** BA networks **c.** RP networks and **d.** SW networks of growing network sizes. Lower values represent more precise quantification of network resilience.



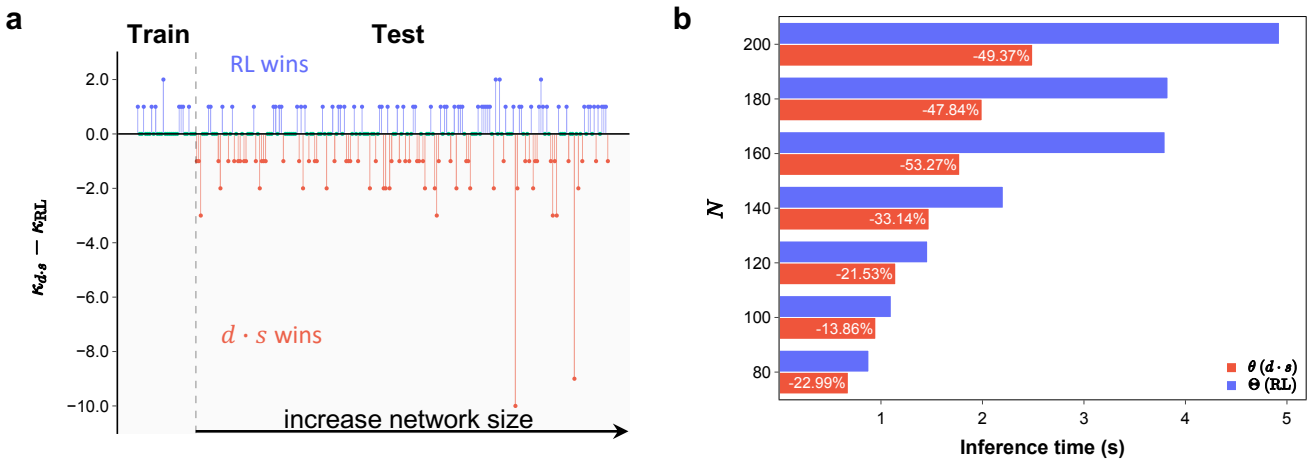
Supplementary Figure 5. The average quantified network resilience across 10 neuronal networks of different methods for **a.** ER networks **b.** BA networks **c.** RP networks and **d.** SW networks of growing network sizes. Lower values represent more precise quantification of network resilience.



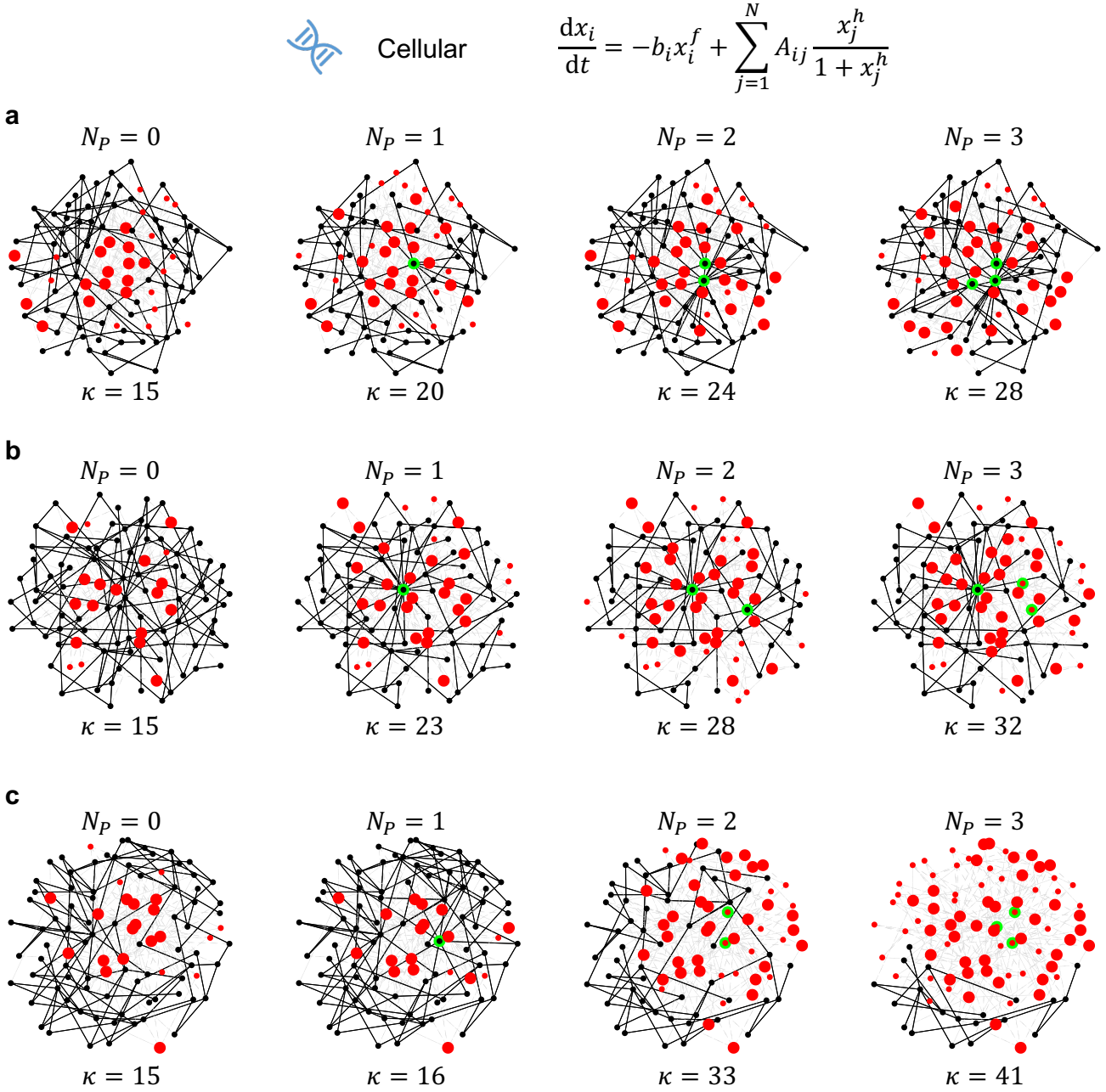
Supplementary Figure 6. **a.** Convergence of the mean estimated resilience by the RL model over 30 training networks (lower means more accurate estimation). The results of three baselines, including degree centrality (DC), resilience centrality (RC) and FINDER, are provided. **b.** Importance value calculated of each node feature by XAI indicating its corresponding contribution to the model prediction. The feature importance values are normalized to $[0, 1]$. **c.** Prediction loss and formula complexity of the obtained equations by SR. The final $d \cdot s$ metric is selected by trading off the two aspects.



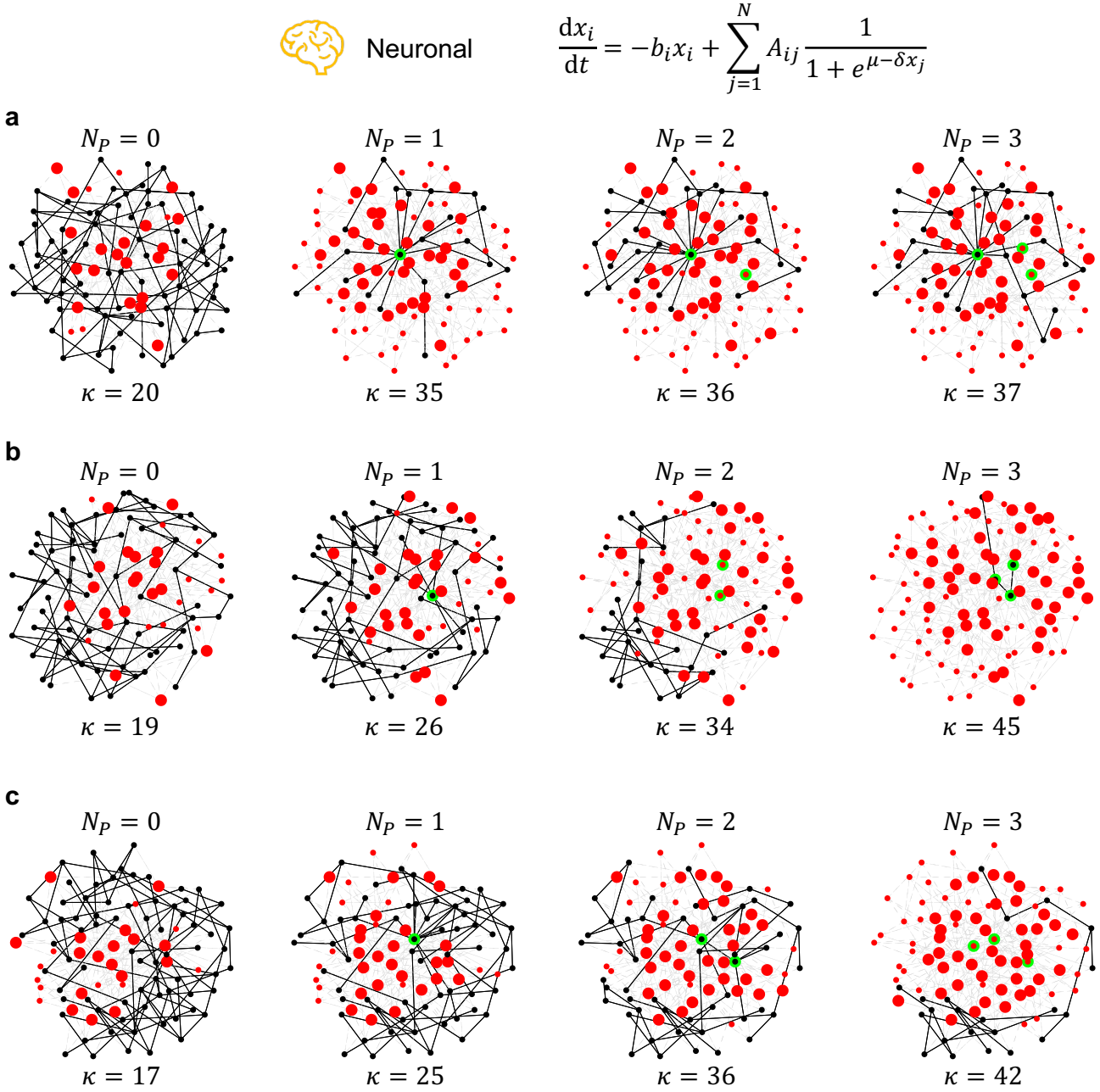
Supplementary Figure 7. **a.** Importance value calculated of each node feature by XAI indicating its corresponding contribution to the prediction of policy network. The feature importance values are normalized to $[0, 1]$. **b.** Importance value calculated of each node feature by XAI indicating its corresponding contribution to the prediction of value network. The feature importance values are normalized to $[0, 1]$.



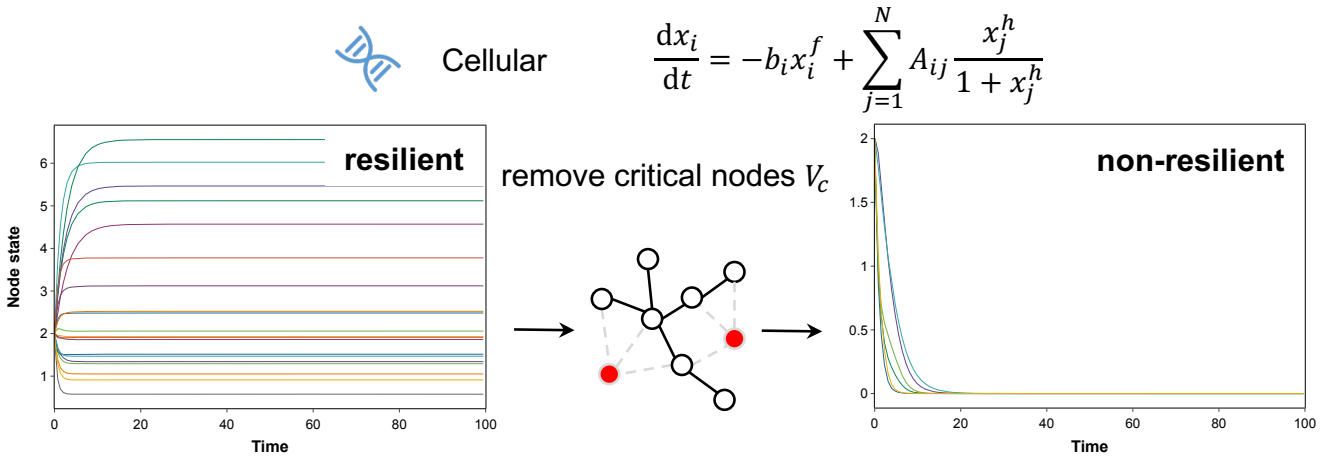
Supplementary Figure 8. **a.** The relative error of the estimated network resilience κ between the $d \cdot s$ formula and the RL model. Experiments are conducted on both the 30 training networks of 80 nodes and 210 unseen test networks ranging from 80 nodes to 200 nodes. Blue and red dots represent that the RL model or the $d \cdot s$ achieves a more precise (lower) quantification of κ, respectively. Green dots indicate a tie. **b.** The average inference time of the RL model Θ and the $d \cdot s$ formula θ across 10 networks of different sizes.



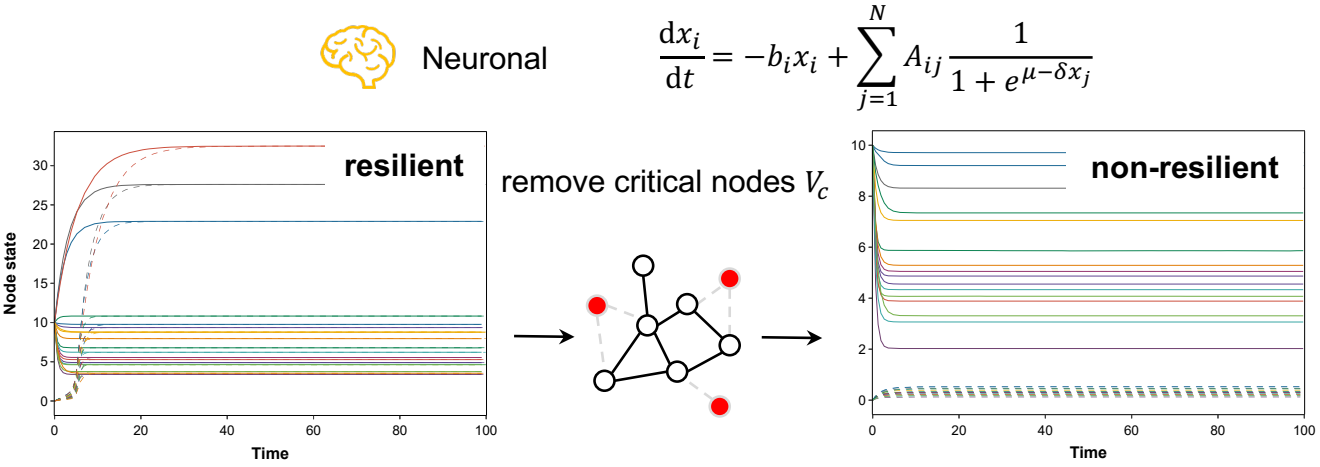
Supplementary Figure 9. The network resilience κ under cellular dynamics when protecting the top N_p nodes indicated by the $d \cdot s$ formula for three synthetic networks of 100 nodes. Nodes with green shells are the safeguarded nodes. Large red nodes are the removed nodes according to $\mathcal{Q} = d \cdot s$ and small red nodes are the discarded nodes that are not in the greatest connected component (GCC) after node removal.



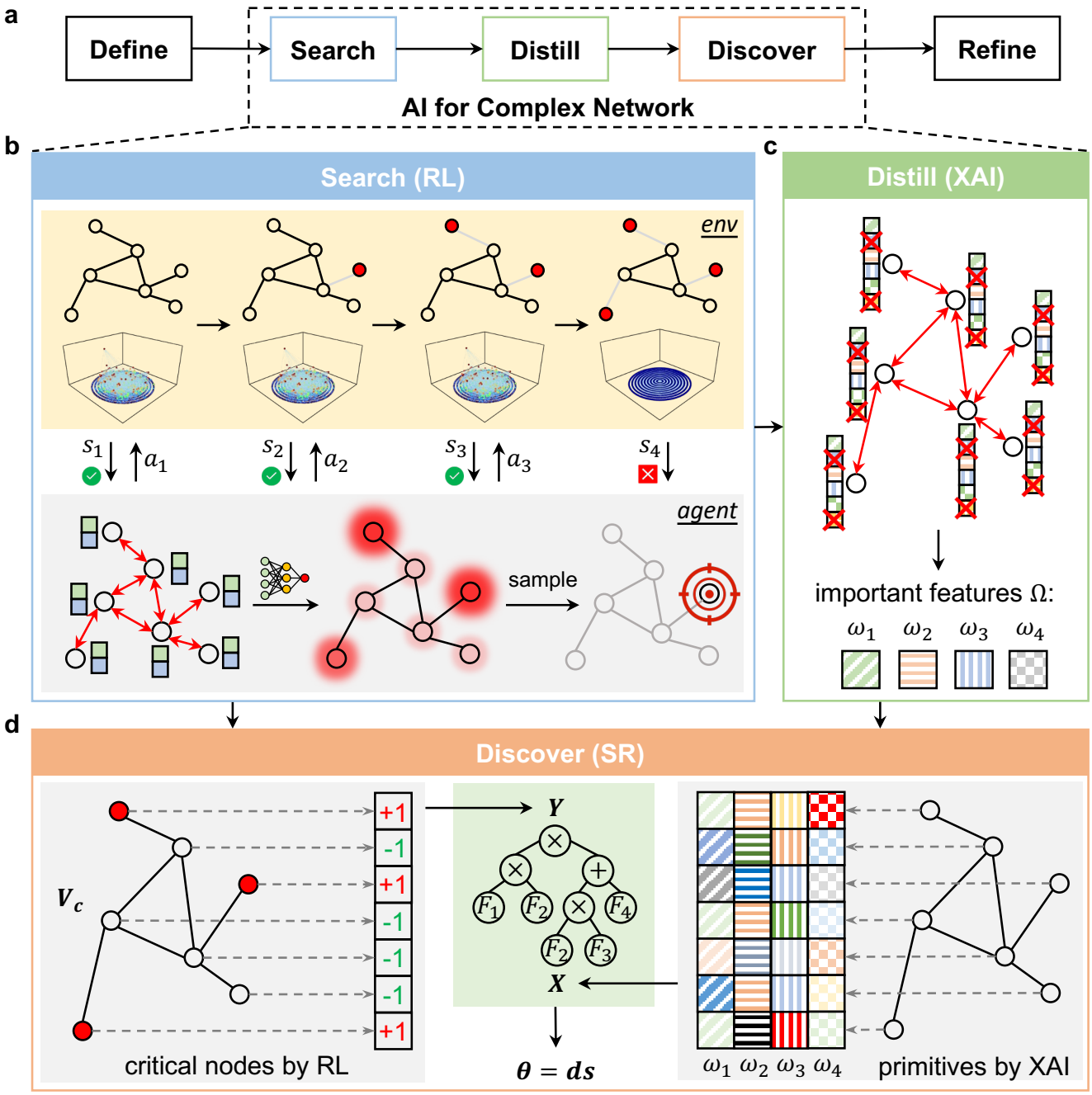
Supplementary Figure 10. The network resilience κ under neuronal dynamics when protecting the top N_p nodes indicated by the $d \cdot s$ formula for three synthetic networks of 100 nodes. Nodes with green shells are the safeguarded nodes. Large red nodes are the removed nodes according to $\mathcal{Q} = d \cdot s$ and small red nodes are the discarded nodes that are not in the greatest connected component (GCC) after node removal.



Supplementary Figure 11. The node state trajectory of a gene regulatory network before and after losing its resilience under node removal, calculated according to the cellular dynamics. The node states are defined as the expression activity of genes, where a resilient network render active node states while a non-resilient network evolves to cell death (inactive genes).



Supplementary Figure 12. The node state trajectory of a brain network before and after losing its resilience under node removal, calculated according to the neuronal dynamics. The node states are defined as the activity of neurons, where a resilient network render the same desired high steady state of different initial conditions, while a non-resilient network evolves to bifurcation of different initial conditions or system inactivity.



Supplementary Figure 13. **a.** The proposed self-inductive AI for complex network framework first solves the complicated problem with AI, then unravels the underlying rules of how AI solves the problem, eventually leading to human-understandable formulas. First, define the problem as a computational manner. Second, search critical nodes of network resilience with an AI model. Third, distill the AI model to identify important node features. Fourth, discover the underlying rules of the AI model that are easy to understand by human. Last, refine the results generated by AI to achieve the final formulaic theory. **b.** An reinforcement learning (RL) model is designed to search critical nodes for network resilience, with one node removed at a step until the network loses its resilience. A graph neural network based model is developed to encode rich node features to representations, which inform node selection. The RL agent interacts with an environment that computes the states of the system, evaluates the resilience of the network, and provides feedback to the agent. **c.** The model prediction is attributed to individual input features, where the contribution of different features are analyzed and a set of important features Ω which dominates the prediction of the RL model are identified. **d.** Empirical expressions between important features by XAI and critical nodes by RL are established. A tangible mathematical formula θ is discovered with symbolic regression indicating the contribution of different nodes to the overall network resilience, which deciphers the intricate mechanisms of the RL model and is more human-understandable.

References

1. Zhang, Y., Shao, C., He, S. & Gao, J. Resilience centrality in complex networks. *Phys. Rev. E* **101**, 022304 (2020).
2. Ren, X.-L., Gleinig, N., Helbing, D. & Antulov-Fantulin, N. Generalized network dismantling. *Proc. national academy sciences* **116**, 6554–6559 (2019).
3. Clusella, P., Grassberger, P., Pérez-Reche, F. J. & Politi, A. Immunization and targeted destruction of networks using explosive percolation. *Phys. review letters* **117**, 208301 (2016).
4. Fan, C., Zeng, L., Sun, Y. & Liu, Y.-Y. Finding key players in complex networks through deep reinforcement learning. *Nat. machine intelligence* **2**, 317–324 (2020).
5. Grassia, M., De Domenico, M. & Mangioni, G. Machine learning dismantling and early-warning signals of disintegration in complex systems. *Nat. Commun.* **12**, 5190 (2021).
6. Sanhedrai, H. *et al.* Reviving a failed network through microscopic interventions. *Nat. Phys.* **18**, 338–349 (2022).
7. Parés, F. *et al.* Fluid communities: A competitive, scalable and diverse community detection algorithm. In *Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications)*, 229–240 (Springer, 2018).
8. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *J. statistical mechanics: theory experiment* **2008**, P10008 (2008).
9. Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T. & Millman, J. (eds.) *Proceedings of the 7th Python in Science Conference*, 11 – 15 (Pasadena, CA USA, 2008).
10. Gao, J., Barzel, B. & Barabási, A.-L. Universal resilience patterns in complex networks. *Nature* **530**, 307–312 (2016).
11. Albert, R., Jeong, H. & Barabási, A.-L. Error and attack tolerance of complex networks. *nature* **406**, 378–382 (2000).
12. Albert, R. & Barabási, A.-L. Statistical mechanics of complex networks. *Rev. modern physics* **74**, 47 (2002).
13. Raffin, A. *et al.* Stable-baselines3: Reliable reinforcement learning implementations. *J. Mach. Learn. Res.* **22**, 1–8 (2021).
14. Ying, Z., Bourgeois, D., You, J., Zitnik, M. & Leskovec, J. Gnnexplainer: Generating explanations for graph neural networks. *Adv. neural information processing systems* **32** (2019).
15. Cranmer, M. Interpretable machine learning for science with pysr and symbolicregression. jl. *arXiv preprint arXiv:2305.01582* (2023).