

RESEARCH ARTICLE

RMut: R package for a Boolean sensitivity analysis against various types of mutations

Hung-Cuong Trinh¹, Yung-Keun Kwon^{2*}**1** Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam, **2** Department of Electrical/Electronic and Computer Engineering, University of Ulsan, Nam-gu, Ulsan, Korea* kwonyk@ulsan.ac.kr

OPEN ACCESS

Citation: Trinh H-C, Kwon Y-K (2019) RMut: R package for a Boolean sensitivity analysis against various types of mutations. PLoS ONE 14(3): e0213736. <https://doi.org/10.1371/journal.pone.0213736>

Editor: José Carlos M. Mombach, Universidade Federal de Santa Maria, BRAZIL

Received: August 18, 2018

Accepted: February 27, 2019

Published: March 19, 2019

Copyright: © 2019 Trinh, Kwon. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the manuscript and its Supporting Information files.

Funding: YKK: This work was supported by National IT Industry Promotion Agency (NIPA) grant funded by the Korea government (MSIP) (S1106-16-1002, Development of smart RMS software for ship maintenance-based fault predictive diagnostics) (<https://www.nipa.kr/eng/main.it>). The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Abstract

There have been many *in silico* studies based on a Boolean network model to investigate network sensitivity against gene or interaction mutations. However, there are no proper tools to examine the network sensitivity against many different types of mutations, including user-defined ones. To address this issue, we developed RMut, which is an R package to analyze the Boolean network-based sensitivity by efficiently employing not only many well-known node-based and edgetic mutations but also novel user-defined mutations. In addition, RMut can specify the mutation area and the duration time for more precise analysis. RMut can be used to analyze large-scale networks because it is implemented in a parallel algorithm using the OpenCL library. In the first case study, we observed that the real biological networks were most sensitive to overexpression/state-flip and edge-addition/-reverse mutations among node-based and edgetic mutations, respectively. In the second case study, we showed that edgetic mutations can predict drug-targets better than node-based mutations. Finally, we examined the network sensitivity to double edge-removal mutations and found an interesting synergistic effect. Taken together, these findings indicate that RMut is a flexible R package to efficiently analyze network sensitivity to various types of mutations. RMut is available at <https://github.com/csclab/RMut>.

Introduction

Many different types of mutations have been used to investigate dynamic behaviors of biological networks; these have focused on essential components identification [1, 2], genetic interactions prediction [3], network intervention [4], and the relationship between dynamic and structural properties [5–7]. In addition, many computational tools have been developed to support *in silico* simulations based on these mutations. For example, CABERNET, a recent Cytoscape app, can assess the dynamics of a network via state-flip, knockout, and overexpression mutations [8]. PANET was developed for parallel analysis of sensitivity-related dynamics against state-flip and rule-flip mutations in large-scale networks [9]. BooleSim [10], Cell Collective [11], and GINsim [12] can manipulate dynamic simulations by employing knockout and overexpression mutations. GDSCalc [13] can evaluate the stability of network dynamics

Competing interests: The authors have declared that no competing interests exist.

upon a state-flip mutation. BoolNet [14] can investigate network sensitivity via state-flip, knockout, and overexpression mutations.

However, each of these tools provides a partial set of previously well-known mutation types, most of which were designed to examine the effects of nodes on network dynamics. On the other hand, there are few tools implementing edgetic mutations, even though recent experimental results have shown that edgetic mutations are useful for genotype-to-phenotype relationship identification and drug discovery [15, 16]. Furthermore, the existing *in silico* tools are not flexible because only a few prespecified mutations can be simulated for analysis. To overcome these limitations, we developed a novel R package called RMut, which can investigate network sensitivity for many well-known node-based and edgetic mutations, as well as user-defined mutations using a synchronous Boolean network model. In addition, we can specify the mutation area and the duration time for more precise analysis. To specify the unknown regulatory rules, we employed the nested canalizing function (NCF) model [17] where a Boolean function is constructed by randomly choosing a sequence of pairs of a canalizing gene and a canalized value. The package provides some additional functions such as attractor identification, feedback/feed-forward search, and centrality calculations. To allow analysis of large-scale networks, we implemented RMut in a parallel computation using the OpenCL library. We note that the core algorithms of RMut were written in Java; thus, a Java SE Development Kit (JDK) is required to run it.

In this study, the usefulness of RMut was demonstrated through three case studies. First, we compared 10 different mutations predefined in RMut over real biological networks, and found that the networks are most sensitive to overexpression/state-flip and edge-addition/-reverse mutations among node-based and edgetic mutations, respectively. In the second case study, we further observed that edgetic mutations can predict drug-targets better than node-based mutations. Interestingly, edge-attenuation (which has never been considered in previous tools) showed high performance in drug-targets prediction. Finally, we examined the network sensitivity to double edge-removal mutations and found a synergistic effect. Altogether, these findings indicate that RMut is a useful and flexible tool for analyzing network dynamics against various types of mutations.

Methods and implementations

This section is organized into four subsections. A Boolean network model employed in this study is first introduced. The next two subsections present predefined mutations that have been widely used in previous studies, and user-defined mutations based on a Java template implementation, respectively. Finally, two network sensitivity measures used in this study are defined.

A Boolean network model

A Boolean network is represented by a directed graph $G(V, E)$, where $V = \{v_1, v_2, \dots, v_N\}$ is a set of nodes, and $E = \{e_1, e_2, \dots, e_M\}$ is a set of ordered pairs of nodes called directed edges. The state of each node $v \in V$ is represented by a Boolean variable having a value of 1 or 0. A directed link $(v_s, v_t) \in E$ has a positive or negative relationship from v_s to v_t (v_s and v_t are called the source and the target nodes of the link, respectively). For a node $v_i \in V$ with d_i incoming links from nodes u_1, u_2, \dots, u_{d_i} where $u_j \neq u_k$ for $\forall j \neq k$, the value of v_i at time $t + 1$ is determined by the values of u_1, u_2, \dots, u_{d_i} at time t by a Boolean function $f_i : \{0, 1\}^{d_i} \rightarrow \{0, 1\}$ (u_1, u_2, \dots, u_{d_i} are called *input* nodes of v_i). Here, we employed a nested canalizing function (NCF) model

[17] to represent the update rule as follows:

$$v_i(t + 1) = f_i(u_1(t), u_2(t), \dots, u_{d_i}(t))$$

$$= \begin{cases} O_1 & \text{if } u_1(t) = I_1 \\ O_2 & \text{if } u_1(t) \neq I_1 \text{ and } u_2(t) = I_2 \\ O_3 & \text{if } u_1(t) \neq I_1 \text{ and } u_2(t) \neq I_2 \text{ and } u_3(t) = I_3 \\ & \vdots \\ O_{d_i} & \text{if } u_1(t) \neq I_1 \text{ and } \dots \text{ and } u_{d_i-1}(t) \neq I_{d_i-1} \text{ and } u_{d_i}(t) = I_{d_i} \\ O_{default_i} & \text{otherwise} \end{cases}$$

where I_k and O_k ($k = 1, 2, \dots, d_i$) are called canalizing and canalized Boolean values, respectively, and $O_{default_i}$ is generally set to $1 - O_{d_i}$. For convenience, we denote f_i as $(I_1, O_1)(I_2, O_2) \dots (I_{d_i}, O_{d_i})O_{default_i}$, which is a sequence of pairs of canalizing and canalized values, followed by the default value. In addition, $v_i(t+1)$ can be expressed in a recursive form of Boolean logic as follows:

$$v_i(t + 1) = f_i(u_1(t), u_2(t), \dots, u_{d_i}(t)) = f_i^{(d_i)}$$

$$\text{where } f_i^{(d_i-k+1)} = \begin{cases} u_k(t) \wedge (f^{(d_i-k)}) & \text{if } k < d_i \text{ and } O_k = 0 \text{ and } O_k = I_k \\ \overline{u_k(t)} \wedge (f^{(d_i-k)}) & \text{if } k < d_i \text{ and } O_k = 0 \text{ and } O_k \neq I_k \\ u_k(t) \vee (f^{(d_i-k)}) & \text{if } k < d_i \text{ and } O_k = 1 \text{ and } O_k = I_k \\ \overline{u_k(t)} \vee (f^{(d_i-k)}) & \text{if } k < d_i \text{ and } O_k = 1 \text{ and } O_k \neq I_k \\ u_k(t) & \text{if } k = d_i \text{ and } O_k = I_k \\ \overline{u_k(t)} & \text{if } k = d_i \text{ and } O_k \neq I_k \end{cases}$$

Herein, $k = 1, 2, \dots, d_i$, and $\overline{u(t)}$ represents the negation value of $u(t)$ (i.e., $\overline{u(t)} = 1 - u(t)$). A canalizing rule means a Boolean rule with a property such that a specific value of one of inputs alone determines the output value. This input and output values are referred to as the canalizing and the canalized value, respectively. A nested canalizing function is a recursive extension of canalizing functions as follows. When the first canalizing variable is not set to the canalizing value, the second canalizing variable and the corresponding canalizing/canalized values are determined. By repeating this determination over all regulatory genes, the nested canalizing function is constructed. It was shown that NCFs properly fit the experimental data obtained from a literature [17]. Furthermore, many logical interaction rules inferred from gene expression data can be represented by NCFs [18, 19]. As in previous studies [17, 20], we independently and randomly specified I_k and O_k values with the probabilities as:

$$Pr(I_k = 1) = \frac{1}{2} \text{ and } Pr(O_k = 1) = \frac{\exp(-2^{-k}\theta)}{1 + \exp(-2^{-k}\theta)} \tag{1}$$

where θ is a constant. In this study, θ was set to 7 following the previous studies [17, 20], which implies that the value of O_k is more likely to be biased to 0 as k decreases. We note that some previous tools such as BoolNet [14] and CABERNET [8] also employed the NCF model to generate random update-rule functions.

A state of a network G is defined as a vector of the states of all nodes, which are synchronously updated by a set of update functions $F = \{f_1, f_2, \dots, f_N\}$. A state trajectory starts from an initial state and eventually converges to either a fixed-point or a limit-cycle attractor. When another trajectory starting from the same initial state along with mutations converges to a different attractor, the network is regarded as sensitive to the mutations.

Predefined mutations

We conducted a survey of different types of mutations in previous *in silico* studies (see Table A in [S1 File](#) for details), and found the 10 most frequently used mutations and classified them into node-based and edgetic mutation groups as follows:

- Node-based mutation group: state-flip, knockout, overexpression, rule-flip, and outcome-shuffle.
- Edgetic mutation group: edge-removal, edge-addition, edge-attenuation, edge-sign-switch, and edge-reverse.

In the following subsections, we explain each mutation in detail, and each is illustrated in [Fig 1](#). Node-based mutations represent changes to most incoming interactions, whereas edgetic mutations represent changes to specific incoming interactions.

Node-based mutations

Let v_i be a node subject to a mutation. Note that each node-based mutation causes a change from $F = \{f_1, \dots, f_i, \dots, f_N\}$ to $F' = \{f_1, \dots, f'_i, \dots, f_N\}$.

- State-flip: This mutation represents a situation in which a protein or gene outputs an opposite state value to expectations [21–24]. More specifically, it can describe a biological process such that genes may become either activated or inhibited by external stimuli such as mutagens [25, 26] and heat stress [27].

Implementation—this mutation is implemented by changing f_i to $f'_i = 1 - f_i$.

- Knockout: This mutation represents the effect of suppressing the expression of a gene or the pharmaceutical inhibition of secondary messenger production or kinase/phosphatase activity [28]. For example, it can be used to simulate the loss of p53 expression in p53 knockout mice in order to reveal a role for p53 in the protection of mice from spontaneous tumorigenesis [29, 30]. The knockout mutation is accomplished through a variety of techniques *in vivo* such as homologous recombination [31–34] and site-specific nucleases [35–37].

Implementation—this mutation is implemented by changing f_i to $f'_i = 0$.

- Overexpression: This mutation represents the effect of induced gene expression [1]. Increased expression of a wild-type gene can also be disruptive to a cell, organism, or phenotypes [38]. The overexpression of HER2, MYC, REL, or AKT2 often drives a variety of human cancers [39]. In addition, the overexpression due to a gene amplification results in drug-, insecticide-, and heavy metal-resistance [40].

Implementation—this mutation is implemented by changing f_i to $f'_i = 1$.

- Rule-flip: This mutation corresponds to a change in the relationships between nodes. The mutation might represent a deleterious change in the function of a protein or gene [41]. It has similar effects to small-scale mutations in the DNA sequence of a gene [42–46].

Implementation—this mutation is implemented by changing f_i to f'_i where every canalyzing and canalyzed value is flipped (i.e., all I_k and O_k changed to $1 - I_k$ and $1 - O_k$, respectively).

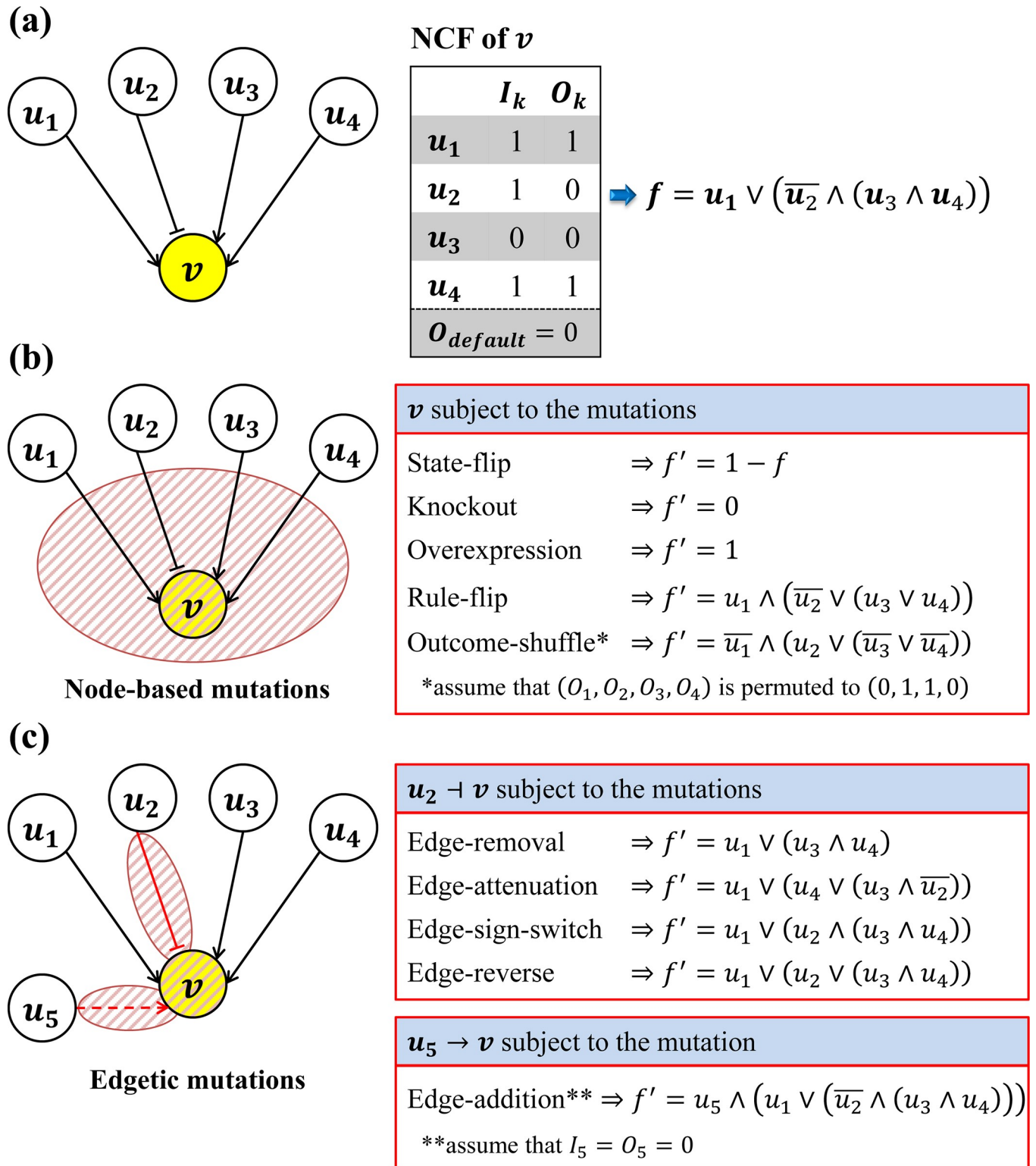


Fig 1. An illustrative example of the predefined mutations implemented in RMut. (a) An example node subject to mutations. Let v be a node with four incoming links from a set of nodes $u_1, u_2, u_3,$ and $u_4,$ and f be the update rule of v . The arrows and bar-headed lines represent positive and negative interactions, respectively. (b) Changes of the update function by node-based mutations subject to node v . The update rule f is modified to f' by each of five node-based mutations. (c) Changes of the update function by edgetic mutations subject to $(u_5, v) \notin E$ or $(u_2, v) \in E$ in the case of the edge-addition and the other edgetic mutations, respectively.

<https://doi.org/10.1371/journal.pone.0213736.g001>

- Outcome-shuffle: This mutation represents the abnormal and chaotic expression of a gene, and may change the function of a protein [47]. Hence, it also has similar effects to small-scale mutations in the DNA sequence of a gene [42–46].

Implementation—this mutation is implemented by changing f_i to f'_i where the analyzed values are permuted (i.e., all analyzed values are randomly changed to $(O'_1, O'_2, \dots, O'_d)$, which is a permutation of (O_1, O_2, \dots, O_d)).

Edgetic mutations

In this study, we define five edgetic mutations. Let v_i be a target node of a link subject to an edgetic mutation, and $f_i = (I_1, O_1) \cdots (I_{d_i}, O_{d_i})O_{default_i}$ is the update rule of v_i . We first explain the edge-addition mutation, and let $(u_k, v_i) \notin E$ be a new edge, which will be added by a mutation.

- Edge-addition: This mutation represents the gain of a new molecular interaction between two proteins or genes [48]. In fact, it is known that some disease potentially can be led through the new incorrect interactions [49, 50]. For instance, the Glu6Val mutation of β -hemoglobin causes sickle cell anemia [51]. Specifically, this mutation introduces a hydrophobic residue to the surface of the protein that can bind to a hydrophobic patch on another hemoglobin molecule under low oxygen conditions, leading to polymerization and the characteristic sickling of the erythrocyte [52].

Implementation—this mutation is implemented by changing E to $E' = E \cup (u_k, v_i)$ and f_i to $f'_i = (I_k, O_k)(I_1, O_1) \cdots (I_{d_i}, O_{d_i})O_{default_i}$. This is accomplished by inserting (I_k, O_k) , which represents the interaction from u_k to v_i , into the first position of f_i . Note that I_k and O_k values are specified by Eq (1).

For the rest of the edgetic mutations, let $(u_j, v_i) \in E$ be an edge subject to a mutation.

- Edge-removal: This mutation represents the loss of a specific molecular interaction between two proteins or genes [48]. For example, the cancer associated C305F missense mutation in the acidic zinc finger domain of Mdm2 results in the loss of Mdm2 binding to L5 and L11. This edgetic perturbation causes the loss of the ribosomal stress response and an increase in c-Myc induced tumorigenesis [53].

Implementation—this mutation is implemented by changing E to $E' = E \setminus (u_j, v_i)$ and f_i to $f'_i = (I_1, O_1) \cdots (I_{j-1}, O_{j-1})(I_{j+1}, O_{j+1}) \cdots (I_{d_i}, O_{d_i})O_{default_i}$ by removing (I_j, O_j) from f_i .

- Edge-attenuation: This mutation corresponds to the weakening of a specific molecular interaction between two proteins or genes [54–56]. As mentioned in a previous study, low-affinity drugs inhibit their targets and can change a strong link into a weak link instead of eliminating the link completely [55].

Implementation—this mutation is implemented by changing f_i to $f'_i = (I_1, O_1) \cdots (I_{j-1}, O_{j-1})(I_{d_i}, O_{d_i})(I_{j+1}, O_{j+1}) \cdots (I_j, O_j)O_{default_i}$ by swapping (I_j, O_j) with (I_{d_i}, O_{d_i}) in f_i . Note that the position order in the sequence of the pairs of analyzed and analyzed values represents the order of precedence in updating the state value of v_i .

- Edge-sign-switch: This mutation represents the switch of the type of molecular interaction between two proteins or genes [57]. An activating interaction is switched to an inhibiting one, and vice versa. This mutation represents the partial change in the function of the source gene/protein as other types of edgetic mutations, but no experimental study about this mutation has been reported.

Implementation—this mutation is implemented by changing f_i to $f'_i = (I_1, O_1) \cdots (1 - I_j, O_j) \cdots (I_{d_i}, O_{d_i}) O_{default_i}$.

- **Edge-reverse:** This mutation represents a switch in both the type and outcome of a molecular interaction or relationship between two proteins or genes [58]. In other words, both the activation/inhibition relationship and the output value of an interaction are switched. As in the edge-sign-switch mutation, there is no experimental study about this mutation yet.

Implementation—this mutation is implemented by changing f_i to $f'_i = (I_1, O_1) \cdots (I_j, 1 - O_j) \cdots (I_{d_i}, O_{d_i}) O_{default_i}$.

These previous mutations were predefined in RMut and therefore can be easily embedded by setting a parameter value. We note that there was no previous tool implementing all of these mutations, and Table 1 shows a comparison of RMut and other tools with respect to available types of mutations.

User-defined mutations

Although most well-known mutations are included in RMut, it is also interesting to simulate new kinds of mutations because some future experimental studies can discover a new mutation type. For example, the edgetic mutations have emerged as a promising strategy for interpretation of genotype-to-phenotype relationships in recent years [50]. To this end, we created a Java template in which a user can flexibly implement novel mutations (Fig 2A). Specifically, RMut provides two functions for node-based and edgetic mutations, respectively. Therefore, many different types of node-based, edgetic mutations, or both can be embedded in RMut. Fig 2B shows an example of an implementation wherein the rule-flip mutation is redefined (see Figures A-I in S1 File for other mutations).

Although it is possible to implement user-defined mutations in some previous tools like BoolNet [14], we note that RMut provides a more systematic way to employ them for the dynamics analysis by using the Java template function. Fig 3 shows an example code for dynamics analysis using a user-defined mutation. As shown in the figure, a user can simply simulate a user-defined mutation, which was saved to a java file, by specifying the file path when calling the sensitivity calculation function ‘calSensitivity()’.

Sensitivity analysis based on a Boolean network model

As described before, we employed a Boolean network model to investigate the network sensitivity. Given a Boolean network, a network state trajectory starting from an initial state

Table 1. Comparisons of RMut and other *in silico* tools based on available mutation types.

Mutation type	RMut	BoolNet	CABERNET	PANET	Cell-Collective	BooleSim	GINsim	GDSCalc
State-flip	✓	✓	✓	✓				✓
Rule-flip	✓			✓				
Knockout & Overexpression	✓	✓	✓		✓	✓	✓	
Outcome-shuffle	✓	✓						
Edge-attenuation	✓							
Edge-removal	✓							
Edge-addition	✓							
Edge-sign-switch	✓							
Edge-reverse	✓	✓						

<https://doi.org/10.1371/journal.pone.0213736.t001>

(a)

```

package mod;
import java.util.ArrayList;
import java.util.concurrent.atomic.AtomicInteger;

public interface Mutation {
    /*
     * input      a sequence of indexes of the input nodes
     * I and O    sequences of analyzing and analyzed values, respectively
     * O_default  the default analyzed value
     */
    void nodeMutation(ArrayList<Integer> input, ArrayList<Integer> I,
                      ArrayList<Integer> O, AtomicInteger O_default);

    /*
     * iSrc      index of the source node
     * tar_input a sequence of indexes of the input nodes of the target node
     * tar_I and tar_O sequences of analyzing and analyzed values of the target
     *           node, respectively
     * tar_O_default the default analyzed value of the target node
     */
    void edgeMutation(int iSrc, ArrayList<Integer> tar_input,
                      ArrayList<Integer> tar_I, ArrayList<Integer> tar_O,
                      AtomicInteger tar_O_default);
}

```

(b)

```

package mod;
import java.util.ArrayList;
import java.util.concurrent.atomic.AtomicInteger;

public class RuleFlip implements Mutation {
    public void nodeMutation(ArrayList<Integer> input, ArrayList<Integer> I,
                             ArrayList<Integer> O, AtomicInteger O_default) {
        int d= input.size(); // indegree of a node subject to a mutation
        if(d > 0) {
            for (int k = 0; k < noInputs; k++) {
                I.set(k, 1 - I.get(k)); // I[k] ← 1 - I[k]
                O.set(k, 1 - O.get(k)); // O[k] ← 1 - O[k]
            }

            O_default.set(1 - O.get(d - 1)); // O_default ← 1 - O[d - 1]
        }
    }

    public void edgeMutation(int iSrc, ArrayList<Integer> tar_input,
                             ArrayList<Integer> tar_I, ArrayList<Integer> tar_O,
                             AtomicInteger tar_O_default) {
        //do nothing
    }
}

```

Fig 2. User-defined mutations in RMut. (a) A Java template for implementation of a user-defined mutation. (b) An example of reimplementing the rule-flip mutation using the template.

<https://doi.org/10.1371/journal.pone.0213736.g002>

converges to an attractor. When another trajectory starting from the same initial state along with a mutation converges to a different attractor, the network is called sensitive to the mutation. To quantify the network sensitivity, we define a mutation group consisting of a subset of nodes, a subset of edges, or both, which describes the area subject to the mutation. This notion allows RMut to use multiple mutations or a single mutation for analysis. We further define the duration time of a mutation, denoted by τ . It means that a mutation is assumed to occur only for a time period $t \in [1, \tau]$. We note that no previous tools incorporated either mutation area or duration time. Given a Boolean network $G(V, E)$, a *network state* at time t can be denoted by an ordered list of state values of all nodes, $\mathbf{v}(t) = [v_1(t), v_2(t), \dots, v_N(t)] \in \{0, 1\}^N$. Every network state

```

# load an example network, the large-scale human signaling network
data(hsn)

# setup OpenCL for parallel computation
setOpencl("gpu")

# generate 1000 random initial-states
states <- generateStates(hsn, 1000)

# generate all possible groups each containing a single node in the HSN network
hsn <- generateGroups(hsn, "all", 1, 0)

# calculate sensitivity values against a user-defined mutation over ten sets of random NCF rules
hsn <- calSensitivity(hsn, states, "D:\\mod\\UserMutation.java", numRuleSets = 10)

# view the calculated sensitivity values and export all results to files
printSensitivity(hsn)
output(hsn)

```

Fig 3. An example of network sensitivity analysis using RMut.

<https://doi.org/10.1371/journal.pone.0213736.g003>

transits to another network state through a set of Boolean update functions $F = \{f_1, f_2, \dots, f_N\}$. We firstly define an attractor more rigorously as follows:

Let $\mathbf{v}(0), \mathbf{v}(1), \dots$, be a network state trajectory starting at $\mathbf{v}(0)$. The attractor is defined as an ordered list of network states $\langle G, F, \mathbf{v}(0) \rangle = [\mathbf{v}(\tau), \mathbf{v}(\tau+1), \dots, \mathbf{v}(\tau+p-1)]$ where τ is the smallest time step such that $\mathbf{v}(t) = \mathbf{v}(t+p)$ for $\forall t \geq \tau$ with $\mathbf{v}(i) \neq \mathbf{v}(j)$ for $\forall i \neq j \in \{\tau, \tau+1, \dots, \tau+p-1\}$ (herein, p is called a length of the attractor).

Generate a set of random initial states S . For each initial state $\mathbf{v}(0) \in S$, we obtain two attractors $\langle G, F, \mathbf{v}(0) \rangle$ and $\langle G', F', \mathbf{v}(0) \rangle$ in the wild-type and the mutant networks, respectively. For convenience, let $\langle G, F, \mathbf{v}(0) \rangle = [\mathbf{v}(\tau), \mathbf{v}(\tau+1), \dots, \mathbf{v}(\tau+p-1)]$ and $\langle G', F', \mathbf{v}(0) \rangle = [\mathbf{v}'(\tau'), \mathbf{v}'(\tau'+1), \dots, \mathbf{v}'(\tau'+p'-1)]$. Finally, we define the network sensitivity as follows:

$$\lambda = \frac{\sum_{\mathbf{v}(0) \in S} d(\langle G, F, \mathbf{v}(0) \rangle, \langle G', F', \mathbf{v}(0) \rangle)}{|S|}, \tag{2}$$

where $d(\cdot)$ denotes a distance function between the wild-type and the mutant attractors. Specifically, we considered two different distance functions based on identicalness and similarity of attractors, respectively, in this study. Let $H(\mathbf{v}(t), \mathbf{v}'(t'))$ the Hamming distance between a pair of Boolean vectors, $\mathbf{v}(t)$ and $\mathbf{v}'(t')$, computed as $\sum_{i=1}^N |v_i(t) - v'_i(t')|$, i.e., the number of different bits. Then the identicalness-based distance is defined as follows:

$$d(\langle G, F, \mathbf{v}(0) \rangle, \langle G', F', \mathbf{v}(0) \rangle) = \begin{cases} 1, & p = p' \text{ and } \exists m \in \{0, \dots, p-1\} \text{ s.t. } \sum_{l=0}^{p-1} H(\mathbf{v}(\tau+l+m), \mathbf{v}'(\tau'+l)) = 0 \\ 0, & \text{otherwise} \end{cases}$$

In addition, the similarity-based distance function is defined considering various possible time lags between two attractors as follows:

$$d(\langle G, F, \mathbf{v}(0) \rangle, \langle G', F', \mathbf{v}(0) \rangle) = \min_{m \in \{0, \dots, d-1\}} \frac{1}{c \cdot N} \sum_{l=0}^{c-1} H(\mathbf{v}(\tau+l+m), \mathbf{v}'(\tau'+l)),$$

where c and d are the least common multiple and the greatest common divisor, respectively, of p and p' . Note that m in both two distance definitions represents a possible time lag between two attractors. As a result, the identicalness-based distance represents whether the wild-type and the mutant attractors are identical to each other or not whereas the similarity-based distance represents the minimum ratio of a bitwise difference between the states sequence in the wild-type and the mutant attractors over the least common period (c) of the two attractors.

With respect to the update function implementation, a user can choose a user-defined NCF or a randomly-generated NCF for each gene, when calling the sensitivity calculation function ‘calSensitivity()’. In the case of the user-defined NCF, a parameter of the file path containing the NCF rules should be specified (see Figure J in S1 File for example). In the case of the random NCF, the sensitivity can be averaged out over a given number of trials which is specified by a parameter ‘numRuleSets’ as shown in Fig 3.

Parallel computation

To allow analysis of large-scale networks, the sensitivity is calculated in parallel using the OpenCL library as in our previous tool-PANET [9]. Specifically, we assign each initial state included in a set of random initial states S in Eq (2) to processing elements of a central

processing unit or graphics processing unit where the wild-type and the mutant attractors are computed in parallel.

Availability

RMut is OS-independent and available at <https://github.com/csclab/RMut>. It requires R 3.5.0 or higher, Java 8 platform (Java SE 8u202 or higher) or Java 11 platform (Java SE 11.0.2 or higher), and OpenCL library (optional). See [S2 File](#) for detailed specification of all functions and [S3 File](#) for user-manual.

All data generated or analyzed during this study are included in this published project and its supplementary information files.

Case studies

To demonstrate the usefulness of RMut, we conducted three case studies using the following real biological networks.

- HSN: This is the large-scale human signaling network (HSN) with 1192 nodes and 3102 links after removing neutral links [59]. Based on the network, some general principles were provided for understanding protein evolution in the context of signaling networks.
- CCSN: This is the canonical cell signaling network (CCSN) with 771 nodes and 1633 links [60]. The network was obtained from <http://stke.sciencemag.org/>, and all the neutral interactions were excluded.
- AMRN: This is the *Arabidopsis* morphogenesis regulatory network (AMRN) with 10 nodes and 22 links [61]. This regulatory network is known to robustly control the process of flower development.

Effects of different types of mutations on network sensitivity

In this case study, we examined the network sensitivity of 10 predefined mutations of RMut. We calculated the average sensitivity value over all nodes or edges in HSN, CCSN, and AMRN assuming that a single mutation occurs at a node (in the case of node-based mutations) or an edge (in the case of edgetic mutations) (Fig 4). The mutation duration time τ was varied from 1 to 10 for the small-scale network, AMRN, and from 2 to 20 by 2 for the large-scale networks, CCSN and HSN.

We note that the sensitivity was averaged over ten sets of random NCF rules. It took about 8.35 days, 3.55 days, and 1.26 minutes to conduct a total of 1000 simulations (10 mutations \times 10 mutation times \times 10 sets of NCF rules) for HSN, CCSN, and AMRN networks, respectively, on a system with an NVIDIA GeForce GTX 680 GPU with 1536 processor at 1 GHz, four-core Intel Core i7-3770 CPU 3.40 GHz, and 8 GB of memory. As shown in Fig 4A–4C, the sensitivity values based on the attractor-identicalness distance increased as the mutation time increased. The sensitivity values based on the attractor-similarity distance also increased against the mutation time on the small-scale network (Fig 4F) but it decreased after some mutation time ($\tau = 8$) for the large-scale networks (Fig 4D and 4E). It turns out that the sensitivity value of some mutations was considerably variable against the mutation time. In addition, we found that the overexpression mutations showed the largest average sensitivity values in all real networks. On the other hand, the knockout mutation showed smaller sensitivity values than the overexpression mutation because the output values of a Boolean variable are likely to be biased to 0 (see Eq (1)). The state-flip mutation showed the second largest sensitivity

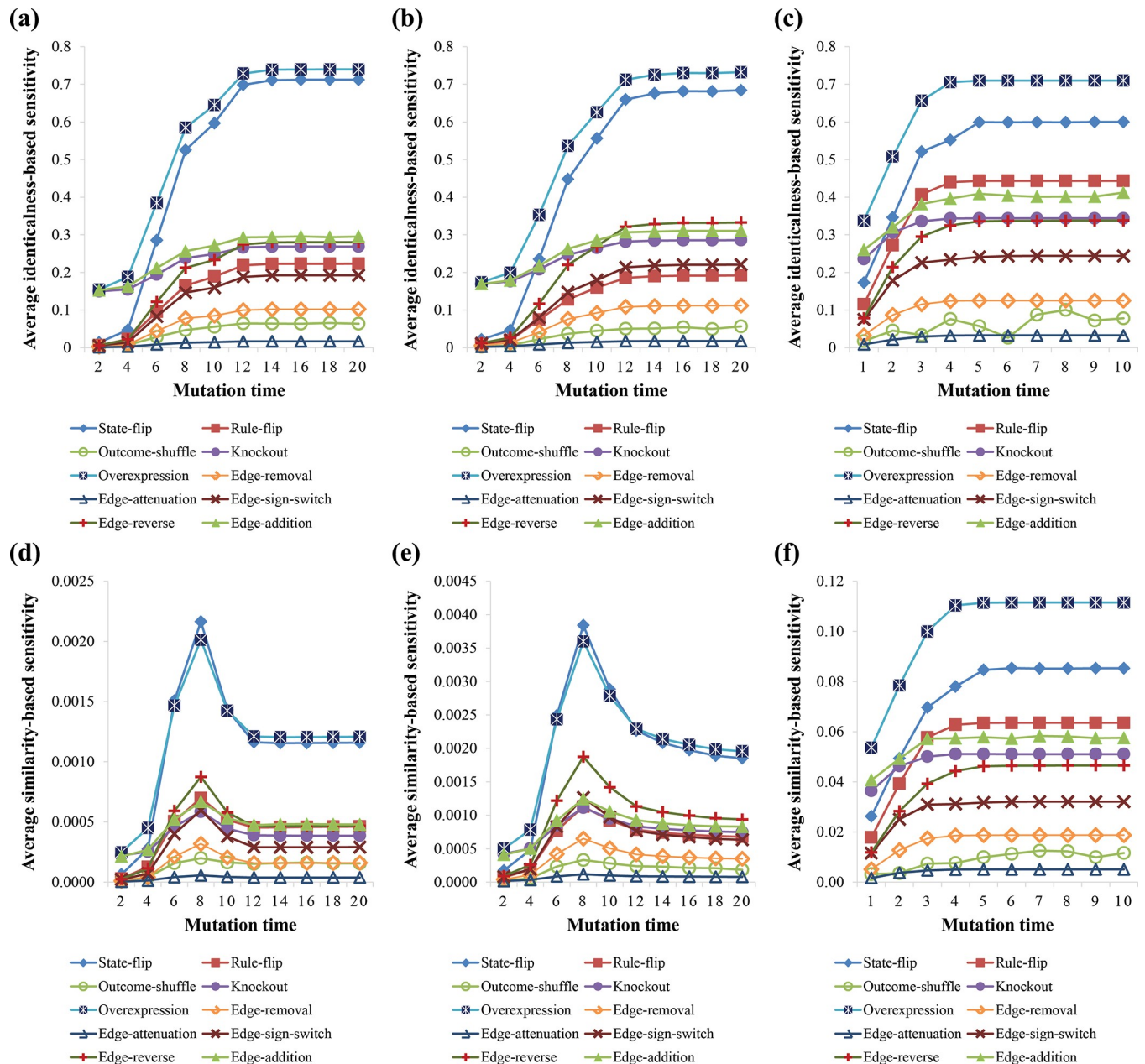


Fig 4. Average sensitivities based on the predefined mutations. (a)-(c) Results of HSN, CCSN, and AMRN networks, respectively, using the identicalness-based sensitivity. (d)-(f) Results of HSN, CCSN, and AMRN networks, respectively, using the similarity-based sensitivity. In each subfigure, Y-axis values represent the average sensitivity values.

<https://doi.org/10.1371/journal.pone.0213736.g004>

values, while the rule-flip showed a moderate level of sensitivity. These node-based mutations have been employed in a variety of studies, but without comparing them with other types of mutations. For instance, knockout and overexpression mutations [1, 28], or rule-flip and state-flip mutations [2, 22] were used to predict essential components in signaling networks. Another study used the knockout mutation to predict mutant phenotypes of fission yeast [62]. Shmulevich, Dougherty [21] developed a method for intervening dynamical network behaviors using the state-flip mutation. Moreover, the relationships between dynamic behaviors and

structural properties were examined based on knockout and rule-flip mutations [5–7]. These previous studies can be extended by employing additional mutations, considering that some recent studies [63–66] have found pleiotropic phenomenon such that different types of mutations can occur in a same gene. Finally, the outcome-shuffle mutation having shown the smallest sensitivity value among the node-based mutations was rarely investigated [14, 47].

Edgetic mutations have been recently considered in experimental studies to better reveal genotype-to-phenotype relationships and drug discovery [15, 16]. Some *in silico* studies have also been conducted [57, 58, 67]. For example, Li, Long [57] investigated the dynamic properties and stability of the yeast cell-cycle network by applying edge-removal, edge-addition, and edge-sign-switch mutations. In this study, we examined a total of five edgetic mutations including all of the previous ones. As shown in Fig 4, we observed that the edge-addition mutations showed much larger sensitivity values than the edge-removal mutations in all real networks. This is interesting because these two mutations seem to be similar to each other in terms of the number of changed interactions. It is known that the edge-addition mutation can lead to disease by incorporating new incorrect interactions [68–70], or can also prevent unexpected network damage by recovering the loss of other interactions [67]. This implies that the edge-addition mutation can be an efficient tool to control the network dynamics, although it is costly and difficult to implement in experimental studies [71]. In addition, edge-reverse mutations showed larger sensitivity values than the edge-sign-switch mutations, which is expected based on the definitions. Previous studies have focused on the former [4, 14, 47, 58, 72] rather than the latter [57]. Specifically, edge-reverse mutations were implemented by flipping an output value of an input variable in a Boolean update rule [4, 58]. Finally, the edge-attenuation mutation, which was not considered in any previous *in silico* studies, showed the smallest sensitivity values in all real networks.

Comparisons of node-based and edgetic sensitivities in drug-targets prediction

To show the usefulness of RMut, we applied it to the drug-target identification problem. A drug-target is a protein, peptide or nucleic acid whose activity can be modulated by a drug to produce a specific effect, which might be a desirable therapeutic effect or an unwanted adverse effect. We hypothesized that a network is susceptible to mutations subject to drug-target genes. To validate this hypothesis, we first profiled a total of 333 drug targets included in HSN using the DrugBank database [73]. We defined the mutation-susceptibility of a node to represent how susceptible a network is to a mutation subject to that node as follows. For a node v and an edge e , let $\lambda(v)$ and $\lambda(e)$ be the network sensitivity value when the mutation group consists only of v and e , respectively. In the case of node-based mutations, the mutation-susceptibility of node v corresponds to $\lambda(v)$, whereas in the case of edgetic mutations, it is specified by $\max_{e \in A} \lambda(e)$, where A is the set of edges that are incident to v . The mutation duration time τ was set to 20 and 8 for identicalness- and similarity-based sensitivities, respectively, because they led to the highest sensitivity to mutations (Fig 4A and 4D). We compared the average mutation-susceptibility values between groups of drug-targets and non-drug targets in HSN (Fig 5A, 5B, 5D and 5E). As shown in the figure, the average mutation-susceptibility values of the drug-target group are significantly higher than those of the non-drug target group (P-values < 0.0001 using t-test) in three or all node-based mutations (Fig 5A and 5D) for identicalness-/similarity-based sensitivity measures, respectively, and in four edgetic mutations for both identicalness-/similarity-based sensitivity measures (Fig 5B and 5E). This implies that edgetic mutations are more stable than node-based mutations in identifying drug targets. For a more precise analysis, we further examined the area under the curve (AUC) when the

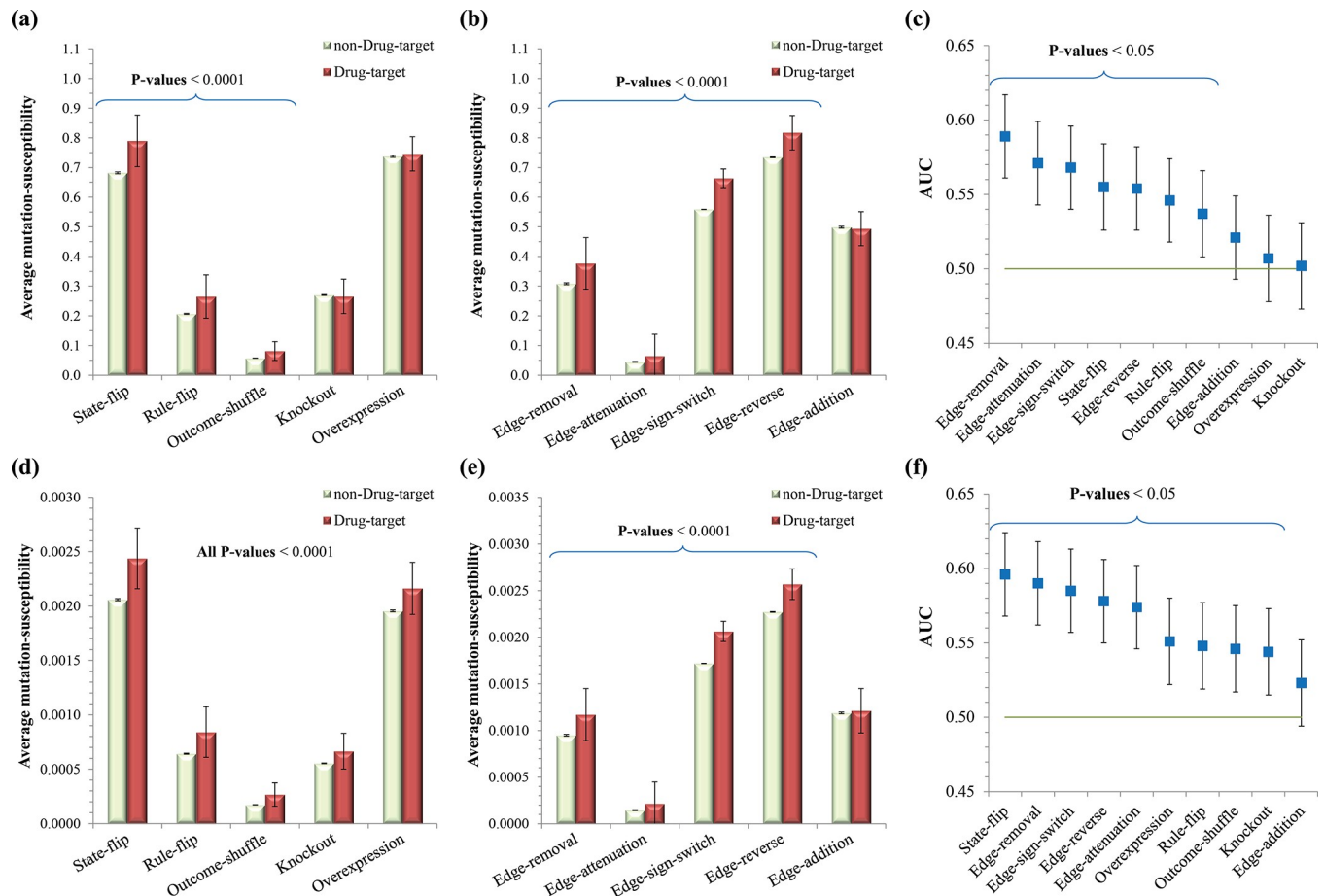


Fig 5. Results of drug-targets prediction based on network sensitivity analysis using RMut. (a)-(b) Comparison of average mutation-susceptibility values between groups of drug-targets and non-drug-targets over node-based and edgetic mutations, respectively, using the identicalness-based sensitivity. (c) AUC values in drug-target prediction using the identicalness-based sensitivity. (d)-(e) Comparison of average mutation-susceptibility values between groups of drug-targets and non-drug-targets over node-based and edgetic mutations, respectively, using the similarity-based sensitivity. (f) AUC values in drug-target prediction using the similarity-based sensitivity. In all sub-figures, the error bars represent 95% confidence intervals.

<https://doi.org/10.1371/journal.pone.0213736.g005>

mutation-susceptibility value was used to prioritize candidate drug-target genes in HSN (Fig 5C and 5F). As shown in the figure, four edgetic mutations, edge-removal, edge-attenuation, edge-sign-switch, and edge-reverse are placed in the top 5 AUC values for both identicalness-/similarity-based sensitivity measures. All of these results illustrate the importance of dynamics analysis based on edgetic mutations, which have not been a primary focus in previous studies compared to node-based mutations.

Synergy effects of double mutant interactions

In genetic interactions, synergy occurs when the contribution of two mutations to the phenotype of a double mutant exceeds the expectations from the additive effects of the individual mutations [74]. Most previous studies have focused on synergy effects based on node-based mutations such as knockout and overexpression mutations [3, 75–78], because they tried to determine the functional roles of a gene or a protein. On the other hand, no previous experimental study focused on the synergistic effects of edgetic mutations. Considering that many experimental results showed the usefulness of edgetic mutations in genotype-to-phenotype relationships and drug discovery [15, 16], *in silico* studies focusing on this issue would be

interesting. Therefore, we conducted a case study to examine the synergy effects based on the edge-removal mutation using RMut. To quantify the synergy effect, we defined the deviation in network sensitivity value by a double edge-removal mutation from the expected one as a single edge-removal mutation as follows:

$$\varepsilon(e_i, e_j) = \lambda(e_i \text{ and } e_j) - \psi(\lambda(e_i), \lambda(e_j)),$$

where $\lambda(e_i \text{ and } e_j)$ and $\lambda(e)$ represent the network sensitivity values (see Eq (2) in Section 2.4) when the mutations group is $\{e_i, e_j\}$ (i.e., a double mutation) and $\{e\}$ (i.e., a single mutation), respectively. In addition, $\psi(x, y)$ represents the expected network sensitivity to a single mutation, and the following three functions were considered for this purpose.

$$\psi_{MAX}(x, y) = \max(x, y) \quad (\text{max})$$

$$\psi_{ROOT}(x, y) = \sqrt{x^2 + y^2} \quad (\text{square root})$$

$$\psi_{ADD}(x, y) = \min(x + y, 1) \quad (\text{additive})$$

Note that $\psi_{MAX} \leq \psi_{ROOT} \leq \psi_{ADD}$ holds. We examined the synergistic effects between all pairs of edges in AMRN and CCSN (Table 2), and we classified (e_i, e_j) into “Synergy” or “No synergy” groups if $\varepsilon(e_i, e_j)$ is greater than a threshold β or not, respectively. The thresholds β were set to 0.1 and 0.01 in the cases of identicalness-based sensitivity (Table 2A) and similarity-based sensitivity (Table 2B), respectively. The mutation duration times were set to 10 for both identicalness-/similarity-based sensitivity measures in the case of AMRN network. For CCSN network, the mutation duration times were set to 20 and 8 for identicalness-based sensitivity

Table 2. Synergistic effects of double edge-removal mutations in two real biological networks, AMRN and CCSN.

(a) Sensitivity based on attractor-identicalness				
AMRN	Synergy		No synergy	
$\psi(x, y)$	Number of edge pairs	%	Number of edge pairs	%
ψ_{MAX}	16	6.93	215	93.07
ψ_{ROOT}	6	2.60	225	97.40
ψ_{ADD}	5	2.16	226	97.84
CCSN	Synergy		No synergy	
$\psi(x, y)$	Number of edge pairs	%	Number of edge pairs	%
ψ_{MAX}	57782	4.34	1274746	95.66
ψ_{ROOT}	10426	0.78	1322102	99.22
ψ_{ADD}	2759	0.21	1329769	99.79
(b) Sensitivity based on attractor-similarity				
AMRN	Synergy		No synergy	
$\psi(x, y)$	Number of edge pairs	%	Number of edge pairs	%
ψ_{MAX}	52	22.51	179	77.49
ψ_{ROOT}	30	12.99	201	87.01
ψ_{ADD}	7	3.03	224	96.97
			s	
CCSN	Synergy		No synergy	
$\psi(x, y)$	Number of edge pairs	%	Number of edge pairs	%
ψ_{MAX}	3255	0.2443	1329273	99.7557
ψ_{ROOT}	3253	0.2441	1329275	99.7559
ψ_{ADD}	3249	0.2438	1329279	99.7562

<https://doi.org/10.1371/journal.pone.0213736.t002>

and similarity-based sensitivity, respectively. These mutation times induced the highest sensitivity to single edge-removal mutations (Fig 4B, 4C, 4E and 4F). As shown in the table, the numbers of synergistic gene pairs were largest and smallest in the case of ψ_{MAX} and ψ_{ADD} , respectively, due to the definitions. In addition, the percentages of the synergistic groups in CCSN were lower than those in AMRN regardless of the type of ψ . This implies that a pair of edges in a large network is less likely to show a synergistic effect than that in a small network. This is because the downstream areas affected by the mutations are not likely to be overlapped in a large network.

In the case of ψ_{ADD} , we further examined the number of edge pairs in the synergy group when both $\lambda(e_i)$ and $\lambda(e_j)$ were nonzero. This condition reflects the situation in which it is most difficult to induce a synergistic effect. Interestingly, we observed a large number of such gene pairs. Specifically, the numbers were 5 out of 5 in AMRN and 242 out of 2759 in CCSN for the identicalness-based sensitivity, and 7 out of 7 in AMRN and 1474 out of 3249 in CCSN for the similarity-based sensitivity.

These edge-pairs tend to maximize the network sensitivity in a synergistic context and could be potential candidates for future experimental studies.

Scalability by parallel computation in RMut

As we mentioned, we implemented RMut in a parallel computation using the OpenCL library. To show the scalability of it, we compared the running times of three versions such as serial, parallel on multi-core CPU, and parallel on GPU modes. We calculated the average sensitivity of the HSN and assumed the knockout mutation. All were tested on a system with an NVIDIA GeForce GTX 680 GPU with 1536 processor at 1 GHz, four-core Intel Core i7-3770 CPU 3.40 GHz, and 8 GB of memory. By varying the number of initial states, we controlled the problem size and Fig 6 shows the result. In the figure, “parallel CPU” and “parallel GPU” represent the results of parallel versions on multi-core CPU and GPU, respectively. As the number of initial-states increases, the speedup by parallelism increases. When the number of initial states was set to 100, the parallel versions were even slower than the serial version. On the other hand, the former versions were about four times faster than the latter. This result explains that the parallel implementation of RMut is properly scalable to the problem size.

Discussion

Although it is ideal to specify all the update rule based on real regulatory relations, most of them are not available, particularly in the case of large-scale biological networks. In this regard, we employed NCFs to randomly specify the update rule in this study. It is known that NCFs can represent various types of regulatory interactions [18, 19, 79–81]. For example, 133 out of 139 rules compiled from a dataset about a transcriptional regulatory network [18] and 39 out of 42 rules inferred from a dataset about signaling pathways [19] were NCFs. Despite these supports, we note that the accurate representation of the regulatory interaction can be limited in our tool. Another issue to be discussed about our Boolean network model is the synchronous update scheme. In fact, it is very likely that the genes in the real signaling networks are regulated in an asynchronous manner. However, it is required to properly specify some unknown parameters to implement the asynchronous scheme such as the number of genes to be updated in a single step and a strategy to choose an update sequence. To avoid this problem, we employed the synchronous update scheme which can be another limitation of our tool.

Next, we discuss the applications of edgetic mutations and the related biological experiments. Despite the experimental cost and the difficulty of implementing the edgetic mutations [82, 83], many recent experimental studies have reported that the edgetic mutations are useful

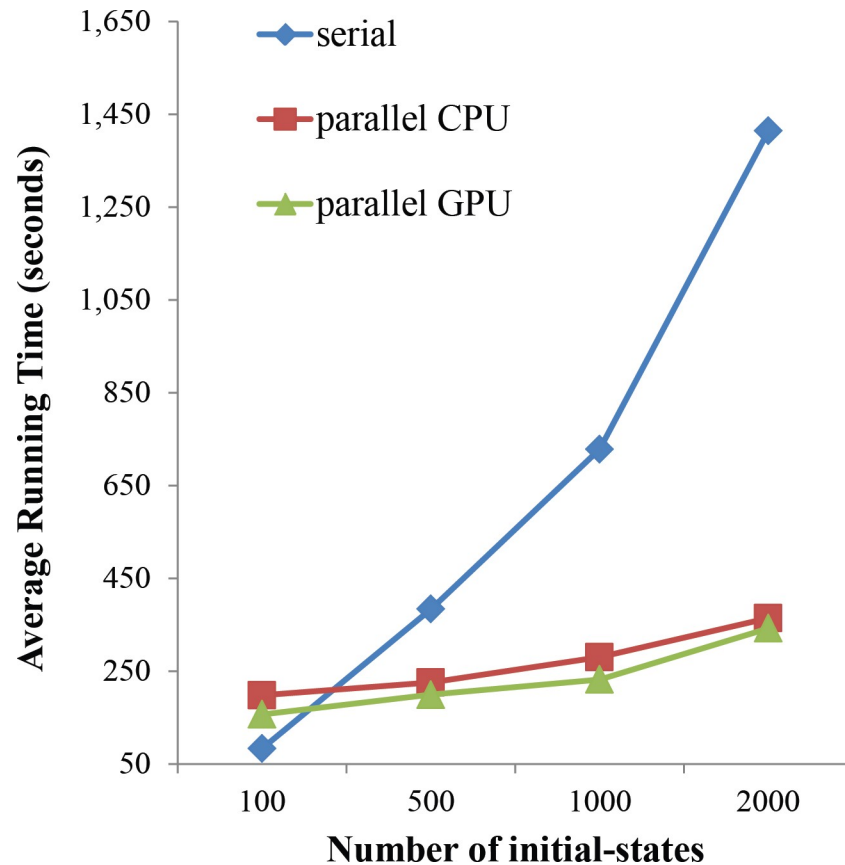


Fig 6. Scalability by parallel computation in RMut. We compared the running times for calculating average sensitivity of the HSN in three modes: serial computation, parallel computation on multi-core CPU (denoted as “parallel CPU”), and parallel computation on GPU (denoted as “parallel GPU”). The knockout mutation is considered and the number of initial states varied from 100 to 2000.

<https://doi.org/10.1371/journal.pone.0213736.g006>

to identify novel genotype-to-phenotype relations [15, 49, 50, 69, 84, 85]. In particular, they mostly employed the edge-removal and the edge-addition mutations. For example, a previous study has shown how much proportion of human diseases may potentially arise from protein-protein interactions-disruptive mutations by using protein structural information [85]. Another study explained that certain PHOX2B variants are associated with neuroblastoma pathogenesis because of their inability to bind to key interacting proteins such as HPCAL1 [84]. It was also shown that the gain of interaction (edge-addition) mutation can lead to a disease by incorporating new incorrect interactions [68–70]. In addition, interaction-targeted drugs called edgetic drugs have been emerged as a novel strategy in the drug discovery [16, 86–89] because they can be more specific than node-targeted ones. Some previous studies focused on small drug-like molecules mediating protein-protein interactions [86–89], and identified a few small molecules through experiments [90–94]. In [90, 91], the authors revealed that Mdm2/Mdm-X interaction is a promising target for therapeutic reactivation of the tumor-suppressor gene p53 in cancer treatments and found small molecule inhibitors to disrupt Mdm2/Mdm-X interaction and activate p53 function. Another study [92] identified ICG-001 which is a small molecule that inhibits the interaction between β -catenin and CBP, and reduces the growth of colon carcinoma cells. On the other hand, there were few experimental studies based on other edgetic mutation types. For example, a previous study found that low-

affinity drugs inhibit their targets and can change the intensity of an interaction instead of eliminating the link completely [55]. We note that this experiment can be simulated by using the edge-attenuation mutation in our tool. Taken together, a variety of edgetic mutations can be prominent in future drug discovery.

Finally, we here summarize the benefits of RMut package over the previous existing tools. The most benefit is that a user can analyze the network dynamics over many different types of mutations as well as novel user-defined ones. In addition, it is possible to more precisely analyze the dynamics by changing various environmental parameters such as the mutation area (i.e., multiple mutations) and the duration time. Moreover, the large-scale networks can be investigated due to the parallel implementation using an OpenCL platform. Our package also features not only the dynamics analysis but also the structure analysis such as calculating node-/edge-based centralities and identifying feedback/feed-forward loops in a single package. Based on these advantages, RMut package can be used in various applications. For example, we can identify some essential components [1, 2] by examining the sensitivity values of the interested components. In addition, it can be used to predict genetic interactions [3] by comparing the sensitivity value of a double gene mutation from the value predicted from single mutations, and reveal the network intervention [4] by applying the state-flip mutation subject to a single gene. It is also possible to investigate an emergent property by examining the relationship between dynamic and structural properties [5–7]. Another application is the drug discovery [15, 16] by computing the sensitivity values of the genes or interactions to identify drug-target candidates. In this way, we believe our tool can give various benefits to many researchers.

Conclusions

We developed RMut, which is an efficient R package to investigate the network sensitivity for both predefined node-based and edgetic mutations. Moreover, new user-defined mutations can be easily embedded using a Java template. RMut also provides more precise analysis by specifying the mutation area and the duration time. We implemented RMut in a parallel algorithm using the OpenCL library to analyze large-scale networks. In this study, we demonstrated the usefulness of RMut through three case studies. First, we compared 10 different mutations predefined in RMut over real biological networks and found that the networks were most sensitive to overexpression/state-flip and edge-addition/-reverse mutations among node-based and edgetic mutations, respectively. In the second case study, we observed that edgetic mutations can predict drug-targets better than node-based mutations. Interestingly, edge-attenuation, which has not been considered in previous tools, showed high performance in drug-target prediction. Finally, we compared double and single edge-removal mutations based on network sensitivity values, and found an interesting synergy effect even for a pair of susceptible edges. Taken together, these findings indicate that RMut can be a useful tool to efficiently analyze network sensitivity against various types of mutations. In future, RMut could be extended to employ arbitrary update-rules or asynchronous update-scheme in the Boolean network model, and provide more visualization features for the analysis.

Supporting information

S1 File. Supporting figures and tables. The file includes the following: Figure A. Defining code of the state-flip mutation. Figure B. Defining code of the knockout mutation. Figure C. Defining code of the overexpression mutation. Figure D. Defining code of the outcome-shuffle mutation. Figure E. Defining code of the edge-removal mutation. Figure F. Defining code of the edge-addition mutation. Figure G. Defining code of the edge-attenuation mutation.

Figure H. Defining code of the edge-sign-switch mutation. Figure I. Defining code of the edge-reverse mutation. Figure J. An example of network sensitivity analysis using user-defined NCF. Table A. List of the well-known mutations and their related studies. (PDF 134 KB) (DOCX)

S2 File. Detailed specification of all functions in RMut. (PDF 129 KB).

(PDF)

S3 File. User manual of RMut. (PDF 1,002 KB).

(PDF)

Acknowledgments

This work was supported by National IT Industry Promotion Agency (NIPA) grant funded by the Korea government (MSIP) (S1106-16-1002, Development of smart RMS software for ship maintenance-based fault predictive diagnostics).

Author Contributions

Conceptualization: Yung-Keun Kwon.

Funding acquisition: Yung-Keun Kwon.

Investigation: Hung-Cuong Trinh.

Methodology: Hung-Cuong Trinh.

Project administration: Yung-Keun Kwon.

Software: Hung-Cuong Trinh.

Supervision: Yung-Keun Kwon.

Writing – original draft: Hung-Cuong Trinh.

Writing – review & editing: Yung-Keun Kwon.

References

1. Azuaje F, Devaux Y, Wagner DR. Identification of potential targets in biological signalling systems through network perturbation analysis. *Biosystems*. 2010; 100(1):55–64. <https://doi.org/10.1016/j.biosystems.2010.01.002> PMID: 20079399
2. Trinh H-C, Kwon Y-K. Effective Boolean dynamics analysis to identify functionally important genes in large-scale signaling networks. *Biosystems*. 2015; 137:64–72. <https://doi.org/10.1016/j.biosystems.2015.07.007> PMID: 26276411
3. Calzone L, Barillot E, Zinovyev A. Predicting genetic interactions from Boolean models of biological networks. *Integrative Biology*. 2015; 7(8):921–9. <https://doi.org/10.1039/c5ib00029g> PMID: 25958956
4. Dehghannasiri R, Yoon BJ, Dougherty ER. Optimal Experimental Design for Gene Regulatory Networks in the Presence of Uncertainty. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. 2015; 12(4):938–50. <https://doi.org/10.1109/TCBB.2014.2377733> PMID: 26357334
5. Kwon Y-K, Kim J, Cho K-H. Dynamical Robustness Against Multiple Mutations in Signaling Networks. *Computational Biology and Bioinformatics*, IEEE/ACM Transactions on. 2015;PP(99). <https://doi.org/10.1109/TCBB.2015.2495251>
6. Hetmanski JHR, Zindy E, Schwartz J-M, Caswell PT. A MAPK-Driven Feedback Loop Suppresses Rac Activity to Promote RhoA-Driven Cancer Cell Invasion. *PLoS Comput Biol*. 2016; 12(5):e1004909. <https://doi.org/10.1371/journal.pcbi.1004909> PMID: 27138333
7. Ortiz-Gutiérrez E, García-Cruz K, Azpeitia E, Castillo A, Sánchez MdIP, Álvarez-Buylla ER. A Dynamic Gene Regulatory Network Model That Recovers the Cyclic Behavior of *Arabidopsis thaliana* Cell Cycle. *PLoS Comput Biol*. 2015; 11(9):e1004486. <https://doi.org/10.1371/journal.pcbi.1004486> PMID: 26340681

8. Paroni A, Graudenzi A, Caravagna G, Damiani C, Mauri G, Antoniotti M. CAbERNET: a Cytoscape app for augmented Boolean models of gene regulatory NETWORKS. *BMC Bioinformatics*. 2016; 17(1):1–12. <https://doi.org/10.1186/s12859-016-0914-z> PMID: 26846964
9. Trinh H-C, Le D-H, Kwon Y-K. PANET: A GPU-Based Tool for Fast Parallel Analysis of Robustness Dynamics and Feed-Forward/Feedback Loop Structures in Large-Scale Biological Networks. *PLoS ONE*. 2014; 9(7):e103010. <https://doi.org/10.1371/journal.pone.0103010> PMID: 25058310
10. Bock M, Scharp T, Talnikar C, Klipp E. BooleSim: an interactive Boolean network simulator. *Bioinformatics*. 2014; 30(1):131–2. <https://doi.org/10.1093/bioinformatics/btt568> PMID: 24078712
11. Helikar T, Kowal B, McClenathan S, Bruckner M, Rowley T, Madrahimov A, et al. The Cell Collective: Toward an open and collaborative approach to systems biology. *BMC Systems Biology*. 2012; 6(1):96.
12. Chaouiya C, Naldi A, Thieffry D. Logical Modelling of Gene Regulatory Networks with GINsim. In: van Helden J, Toussaint A, Thieffry D, editors. *Bacterial Molecular Networks: Methods and Protocols*. New York, NY: Springer New York; 2012. p. 463–79.
13. Elmeligy Abdelhamid SH, Kuhlman CJ, Marathe MV, Mortveit HS, Ravi SS. GDSCalc: A Web-Based Application for Evaluating Discrete Graph Dynamical Systems. *PLoS ONE*. 2015; 10(8):e0133660. <https://doi.org/10.1371/journal.pone.0133660> PMID: 26263006
14. Müssel C, Hopfensitz M, Kestler HA. BoolNet—an R package for generation, reconstruction and analysis of Boolean networks. *Bioinformatics*. 2010; 26(10):1378–80. <https://doi.org/10.1093/bioinformatics/btq124> PMID: 20378558
15. Sahni N, Yi S, Taipale M, Fuxman Bass Juan I, Coulombe-Huntington J, Yang F, et al. Widespread Macromolecular Interaction Perturbations in Human Genetic Disorders. *Cell*. 2015; 161(3):647–60. <https://doi.org/10.1016/j.cell.2015.04.013> PMID: 25910212
16. Pardo EP, Godzik A. Analysis of Individual Protein Regions Provides Novel Insights on Cancer Pharmacogenomics. *PLoS Comput Biol*. 2015; 11(1):e1004024. <https://doi.org/10.1371/journal.pcbi.1004024> PMID: 25568936
17. Kauffman S, Peterson C, Samuelsson B, Troein C. Random Boolean network models and the yeast transcriptional network. *Proceedings of the National Academy of Sciences*. 2003; 100(25):14796–9.
18. Harris SE, Sawhill BK, Wuensche A, Kauffman S. A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity*. 2002; 7(4):23–40. <https://doi.org/10.1002/cplx.10022>
19. Naldi A, Carneiro J, Chaouiya C, Thieffry D. Diversity and Plasticity of Th Cell Types Predicted from Regulatory Network Modelling. *PLoS Comput Biol*. 2010; 6(9):e1000912. <https://doi.org/10.1371/journal.pcbi.1000912> PMID: 20824124
20. Kauffman S, Peterson C, Samuelsson B, Troein C. Genetic networks with canalizing Boolean rules are always stable. *Proceedings of the National Academy of Sciences of the United States of America*. 2004; 101(49):17102–7. <https://doi.org/10.1073/pnas.0407783101> PMID: 15572453
21. Shmulevich I, Dougherty ER, Zhang W. Gene perturbation and intervention in probabilistic Boolean networks. *Bioinformatics*. 2002; 18(10):1319–31. PMID: 12376376
22. Kwon Y-K, Choi S, Cho K-H. Investigations into the relationship between feedback loops and functional importance of a signal transduction network based on Boolean network modeling. *BMC Bioinformatics*. 2007; 8(1):384. <https://doi.org/10.1186/1471-2105-8-384> PMID: 17935633
23. Davidich MI, Bornholdt S. Boolean Network Model Predicts Cell Cycle Sequence of Fission Yeast. *PLoS ONE*. 2008; 3(2):e1672. <https://doi.org/10.1371/journal.pone.0001672> PMID: 18301750
24. Álvarez-Buylla ER, Chaos Á, Aldana M, Benítez M, Cortes-Poza Y, Espinosa-Soto C, et al. Floral Morphogenesis: Stochastic Explorations of a Gene Network Epigenetic Landscape. *PLoS ONE*. 2008; 3(11):e3626. <https://doi.org/10.1371/journal.pone.0003626> PMID: 18978941
25. Tillich UM, Lehmann S, Schulze K, Dühning U, Frohme M. The Optimal Mutagen Dosage to Induce Point-Mutations in *Synechocystis* sp. PCC6803 and Its Application to Promote Temperature Tolerance. *PLoS ONE*. 2012; 7(11):e49467. <https://doi.org/10.1371/journal.pone.0049467> PMID: 23185339
26. Parry JM. The use of yeast cultures for the detection of environmental mutagens using a fluctuation test. *Mutation Research/Environmental Mutagenesis and Related Subjects*. 1977; 46(3):165–75. [https://doi.org/10.1016/0165-1161\(77\)90023-1](https://doi.org/10.1016/0165-1161(77)90023-1).
27. Suzuki N, Bassil E, Hamilton JS, Inupakutika MA, Zandalinas SI, Tripathy D, et al. ABA Is Required for Plant Acclimation to a Combination of Salt and Heat Stress. *PLoS ONE*. 2016; 11(1):e0147625. <https://doi.org/10.1371/journal.pone.0147625> PMID: 26824246
28. Li S, Assmann SM, Albert R. Predicting Essential Components of Signal Transduction Networks: A Dynamic Model of Guard Cell Abscisic Acid Signaling. *PLoS Biol*. 2006; 4(10):e312. <https://doi.org/10.1371/journal.pbio.0040312> PMID: 16968132

29. Dudgeon C, Chan C, Kang W, Sun Y, Emerson R, Robins H, et al. The evolution of thymic lymphomas in p53 knockout mice. *Genes & Development*. 2014; 28(23):2613–20.
30. Donehower LA, Harvey M, Vogel H, McArthur MJ, Montgomery CA, Park SH, et al. Effects of genetic background on tumorigenesis in p53-deficient mice. *Molecular Carcinogenesis*. 1995; 14(1):16–22. <https://doi.org/10.1002/mc.2940140105> PMID: 7546219
31. Strepp R, Scholz S, Kruse S, Speth V, Reski R. Plant nuclear gene knockout reveals a role in plastid division for the homolog of the bacterial cell division protein FtsZ, an ancestral tubulin. *Proceedings of the National Academy of Sciences*. 1998; 95(8):4368.
32. Ni W, Qiao J, Hu S, Zhao X, Regouski M, Yang M, et al. Efficient Gene Knockout in Goats Using CRISPR/Cas9 System. *PLoS ONE*. 2014; 9(9):e106718. <https://doi.org/10.1371/journal.pone.0106718> PMID: 25188313
33. Dimitrov L, Pedersen D, Ching KH, Yi H, Collarini EJ, Izquierdo S, et al. Germline Gene Editing in Chickens by Efficient CRISPR-Mediated Homologous Recombination in Primordial Germ Cells. *PLoS ONE*. 2016; 11(4):e0154303. <https://doi.org/10.1371/journal.pone.0154303> PMID: 27099923
34. Tong C, Li P, Wu NL, Yan Y, Ying Q-L. Production of p53 gene knockout rats by homologous recombination in embryonic stem cells. *Nature*. 2010; 467:211. <https://doi.org/10.1038/nature09368> <https://www.nature.com/articles/nature09368#supplementary-information> PMID: 20703227
35. Hammond A, Galizi R, Kyrou K, Simoni A, Siniscalchi C, Katsanos D, et al. A CRISPR-Cas9 gene drive system targeting female reproduction in the malaria mosquito vector *Anopheles gambiae*. *Nature Biotechnology*. 2015; 34:78. <https://doi.org/10.1038/nbt.3439> <https://www.nature.com/articles/nbt.3439#supplementary-information> PMID: 26641531
36. Carlson DF, Tan W, Lillico SG, Stverakova D, Proudfoot C, Christian M, et al. Efficient TALEN-mediated gene knockout in livestock. *Proceedings of the National Academy of Sciences*. 2012.
37. Urnov FD, Rebar EJ, Holmes MC, Zhang HS, Gregory PD. Genome editing with engineered zinc finger nucleases. *Nature Reviews Genetics*. 2010; 11:636. <https://doi.org/10.1038/nrg2842> PMID: 20717154
38. Prelich G. Gene Overexpression: Uses, Mechanisms, and Interpretation. *Genetics*. 2012; 190(3):841. <https://doi.org/10.1534/genetics.111.136911> PMID: 22419077
39. Shastry BS. Overexpression of genes in health and sickness. A bird's eye view. *Comparative Biochemistry and Physiology Part B: Biochemistry and Molecular Biology*. 1995; 112(1):1–13. [https://doi.org/10.1016/0305-0491\(95\)00055-D](https://doi.org/10.1016/0305-0491(95)00055-D).
40. Stark GR, Wahl GM. Gene Amplification. *Annual Review of Biochemistry*. 1984; 53(1):447–91. <https://doi.org/10.1146/annurev.bi.53.070184.002311> PMID: 6383198
41. Ng PC, Henikoff S. SIFT: predicting amino acid changes that affect protein function. *Nucleic Acids Research*. 2003; 31(13):3812–4. <https://doi.org/10.1093/nar/gkg509> PMID: 12824425
42. Dorshorst B, Henegar C, Liao X, Sällman Almén M, Rubin C-J, Ito S, et al. Dominant Red Coat Color in Holstein Cattle Is Associated with a Missense Mutation in the Coatmer Protein Complex, Subunit Alpha (COPA) Gene. *PLoS ONE*. 2015; 10(6):e0128969. <https://doi.org/10.1371/journal.pone.0128969> PMID: 26042826
43. Kamburov A, Lawrence MS, Polak P, Leshchiner I, Lage K, Golub TR, et al. Comprehensive assessment of cancer missense mutation clustering in protein structures. *Proceedings of the National Academy of Sciences*. 2015; 112(40):E5486.
44. Styrkarsdottir U, Thorleifsson G, Sulem P, Gudbjartsson DF, Sigurdsson A, Jonasdottir A, et al. Nonsense mutation in the LGR4 gene is associated with several human diseases and other traits. *Nature*. 2013; 497:517. <https://doi.org/10.1038/nature12124> <https://www.nature.com/articles/nature12124#supplementary-information> PMID: 23644456
45. Downs LM, Wallin-Håkansson B, Bournsnell M, Marklund S, Hedhammar Å, Truvé K, et al. A Frameshift Mutation in Golden Retriever Dogs with Progressive Retinal Atrophy Endorses SLC4A3 as a Candidate Gene for Human Retinal Degenerations. *PLoS ONE*. 2011; 6(6):e21452. <https://doi.org/10.1371/journal.pone.0021452> PMID: 21738669
46. Ogura Y, Bonen DK, Inohara N, Nicolae DL, Chen FF, Ramos R, et al. A frameshift mutation in NOD2 associated with susceptibility to Crohn's disease. *Nature*. 2001; 411:603. <https://doi.org/10.1038/35079114> PMID: 11385577
47. Xiaoning Q, Dougherty ER. Effect of Function Perturbation on the Steady-State Distribution of Genetic Regulatory Networks: Optimal Structural Intervention. *Signal Processing, IEEE Transactions on*. 2008; 56(10):4966–76. <https://doi.org/10.1109/tsp.2008.928089>
48. Zhong Q, Simonis N, Li QR, Charlotheaux B, Heuze F, Klitgord N, et al. Edgetic perturbation models of human inherited disorders. *Molecular Systems Biology*. 2009; 5(1). <https://doi.org/10.1038/msb.2009.80> PMID: 19888216

49. Yates CM, Sternberg MJE. The Effects of Non-Synonymous Single Nucleotide Polymorphisms (nsSNPs) on Protein–Protein Interactions. *Journal of Molecular Biology*. 2013; 425(21):3949–63. <https://doi.org/10.1016/j.jmb.2013.07.012> PMID: 23867278
50. Sahni N, Yi S, Zhong Q, Jaikhan N, Charlotiaux B, Cusick ME, et al. Edgotype: a fundamental link between genotype and phenotype. *Current Opinion in Genetics & Development*. 2013; 23(6):649–57. <https://doi.org/10.1016/j.gde.2013.11.002>.
51. Ingram VM. A Specific Chemical Difference Between the Globins of Normal Human and Sickle-Cell Anæmia Hæmoglobin. *Nature*. 1956; 178:792. <https://doi.org/10.1038/178792a0> PMID: 13369537
52. Wishner BC, Ward KB, Lattman EE, Love WE. Crystal structure of sickle-cell deoxyhemoglobin at 5 Å resolution. *Journal of Molecular Biology*. 1975; 98(1):179–94. [https://doi.org/10.1016/S0022-2836\(75\)80108-2](https://doi.org/10.1016/S0022-2836(75)80108-2). PMID: 1195378
53. Macias E, Jin A, Deisenroth C, Bhat K, Mao H, Lindström MS, et al. An ARF-Independent c-MYC-Activated Tumor Suppression Pathway Mediated by Ribosomal Protein-Mdm2 Interaction. *Cancer Cell*. 2010; 18(3):231–43. <https://doi.org/10.1016/j.ccr.2010.08.007> PMID: 20832751
54. Grindrod P, Higham DJ. A dynamical systems view of network centrality. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. 2014; 470(2165).
55. Csermely P, Ágoston V, Pongor S. The efficiency of multi-target drugs: the network approach might help drug design. *Trends in Pharmacological Sciences*. 2005; 26(4):178–82. <https://doi.org/10.1016/j.tips.2005.02.007> PMID: 15808341
56. Ágoston V, Csermely P, Pongor S. Multiple, weak hits confuse complex systems: A transcriptional regulatory network as an example. *Physical Review E*. 2005; 71(5):051909. [citeulike-article-id:8954843](https://doi.org/10.1103/PhysRevE.71.051909).
57. Li F, Long T, Lu Y, Ouyang Q, Tang C. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*. 2004; 101(14):4781–6. <https://doi.org/10.1073/pnas.0305937101> PMID: 15037758
58. Hu M, Shen L, Zan X, Shang X, Liu W. An efficient algorithm to identify the optimal one-bit perturbation based on the basin-of-state size of Boolean networks. *Scientific Reports*. 2016; 6:26247. <https://doi.org/10.1038/srep26247> PMC4872544. PMID: 27196530
59. Cui Q, Purisima E, Wang E. Protein evolution on a human signaling network. *BMC Systems Biology*. 2009; 3(1):21. <https://doi.org/10.1186/1752-0509-3-21> PMID: 19226461
60. Kwon Y-K, Cho K-H. Coherent coupling of feedback loops: a design principle of cell signaling networks. *Bioinformatics*. 2008; 24(17):1926–32. <https://doi.org/10.1093/bioinformatics/btn337> PMID: 18596076
61. Mendoza L, Thieffry D, Alvarez-Buylla ER. Genetic control of flower morphogenesis in *Arabidopsis thaliana*: a logical analysis. *Bioinformatics*. 1999; 15(7):593–606. <https://doi.org/10.1093/bioinformatics/15.7.593>
62. Davidich MI, Bornholdt S. Boolean Network Model Predicts Knockout Mutant Phenotypes of Fission Yeast. *PLoS ONE*. 2013; 8(9):e71786. <https://doi.org/10.1371/journal.pone.0071786> PMID: 24069138
63. Milani D, Pezzani L, Tabano S, Miozzo M. Beckwith-Wiedemann and IMAGE syndromes: two very different diseases caused by mutations on the same gene. *The application of clinical genetics*. 2014; 7:169–75. <https://doi.org/10.2147/TACG.S35474> PMID: 25258553.
64. Banjar H, Moghrabi N, Alotaibi T, Alotaibi S, Gamalmaz H. The first case report of double homozygous of 2 different mutations in the CFTR gene in Saudi Arabia. *International Journal of Pediatrics and Adolescent Medicine*. 2017; 4(1):41–6. <https://doi.org/10.1016/j.ijpam.2016.09.001> PMID: 30805499
65. Derks MFL, Herrero-Medrano JM, Crooijmans RPMA, Vereijken A, Long JA, Megens H-J, et al. Early and late feathering in turkey and chicken: same gene but different mutations. *Genetics Selection Evolution*. 2018; 50(1):7. <https://doi.org/10.1186/s12711-018-0380-3> PMID: 29566646
66. Horani A, Huang T, Brody SL. Different Mutations in Cilia Preassembly Gene HEATR2 Are Associated with Variable Clinical Phenotypes and Defects in Proteostasis. C18 BEST OF PEDIATRICS. *American Thoracic Society International Conference Abstracts: American Thoracic Society*; 2018. p. A4467-A.
67. Campbell C, Albert R. Stabilization of perturbed Boolean network attractors through compensatory interactions. *BMC Systems Biology*. 2014; 8(1):1–16. <https://doi.org/10.1186/1752-0509-8-53> PMID: 24885780
68. Vogt G, Vogt B, Chuzhanova N, Julenius K, Cooper DN, Casanova J-L. Gain-of-glycosylation mutations. *Current Opinion in Genetics & Development*. 2007; 17(3):245–51. <http://dx.doi.org/10.1016/j.gde.2007.04.008>.
69. Ingram VM. A Specific Chemical Difference Between the Globins of Normal Human and Sickle-Cell Anaemia Haemoglobin. *Nature*. 1956; 178(4537):792–4. PMID: 13369537
70. van Oijen MGCT, Slootweg PJ. Gain-of-Function Mutations in the Tumor Suppressor Gene p53. *American Association for Cancer Research*. 2000; 6(6):2138–45.

71. Chan H, Akoglu L. Optimizing network robustness by edge rewiring: a general framework. *Data Mining and Knowledge Discovery*. 2016;1–31. <https://doi.org/10.1007/s10618-016-0455-0>
72. Xiao Y, Dougherty ER. The impact of function perturbations in Boolean networks. *Bioinformatics*. 2007; 23(10):1265–73. <https://doi.org/10.1093/bioinformatics/btm093> PMID: 17379691
73. Law V, Knox C, Djoumbou Y, Jewison T, Guo AC, Liu Y, et al. DrugBank 4.0: shedding new light on drug metabolism. *Nucleic Acids Research*. 2014; 42(D1):D1091–D7.
74. Pérez-Pérez JM, Candela H, Micol JL. Understanding synergy in genetic interactions. *Trends in Genetics*. 2009; 25(8):368–76. <https://doi.org/10.1016/j.tig.2009.06.004> PMID: 19665253
75. Drees BL, Thorsson V, Carter GW, Rives AW, Raymond MZ, Avila-Campillo I, et al. Derivation of genetic interaction networks from quantitative phenotype data. *Genome Biology*. 2005; 6(4):R38–R. <https://doi.org/10.1186/gb-2005-6-4-r38> PMC1088966. PMID: 15833125
76. Segre D, DeLuna A, Church GM, Kishony R. Modular epistasis in yeast metabolism. *Nat Genet*. 2005; 37(1):77–83. <https://doi.org/10.1038/ng1489> PMID: 15592468
77. Snitkin ES, Segrè D. Epistatic Interaction Maps Relative to Multiple Metabolic Phenotypes. *PLOS Genetics*. 2011; 7(2):e1001294. <https://doi.org/10.1371/journal.pgen.1001294> PMID: 21347328
78. Alam MT, Zeleznik A, Mülleder M, Shliaha P, Schwarz R, Capuano F, et al. The metabolic background is a global player in *Saccharomyces* gene expression epistasis. *Nature Microbiology*. 2016; 1:15030. <https://doi.org/10.1038/nmicrobiol.2015.30> <http://www.nature.com/articles/nmicrobiol201530#supplementary-information>. PMID: 27572163
79. Murrugarra D, Laubenbacher R. Regulatory patterns in molecular interaction networks. *Journal of Theoretical Biology*. 2011; 288(0):66–72. <https://doi.org/10.1016/j.jtbi.2011.08.015> PMID: 21872607
80. Faure A, Naldi A, Chaouiya C, Thieffry D. Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle. *Bioinformatics*. 2006; 22(14):e124–31. <https://doi.org/10.1093/bioinformatics/btl210> PMID: 16873462
81. González A, Chaouiya C, Thieffry D. Dynamical Analysis of the Regulatory Network Defining the Dorsal–Ventral Boundary of the *Drosophila* Wing Imaginal Disc. *Genetics*. 2006; 174(3):1625–34. <https://doi.org/10.1534/genetics.106.061218> PMID: 16951066
82. Domingo G-R, Holger G. Targeting Protein-Protein Interactions with Small Molecules: Challenges and Perspectives for computational Binding Epitope Detection and Ligand Finding. *Current Medicinal Chemistry*. 2006; 13(22):2607–25. <http://dx.doi.org/10.2174/092986706778201530>. PMID: 17017914
83. Laraia L, McKenzie G, Spring Davi R, Venkataraman Ashok R, Huggins David J. Overcoming Chemical, Biological, and Computational Challenges in the Development of Inhibitors Targeting Protein-Protein Interactions. *Chemistry & Biology*. 2015; 22(6):689–703. <https://doi.org/10.1016/j.chembiol.2015.04.019>.
84. Wang W, Zhong Q, Teng L, Bhatnagar N, Sharma B, Zhang X, et al. Mutations that disrupt PHOXB interaction with the neuronal calcium sensor HPCAL1 impede cellular differentiation in neuroblastoma. *Oncogene*. 2014; 33(25):3316–24. <https://doi.org/10.1038/ncr.2013.290> PMID: 23873030
85. Wang X, Wei X, Thijssen B, Das J, Lipkin SM, Yu H. Three-dimensional reconstruction of protein networks provides insight into human genetic disease. *Nat Biotech*. 2012; 30(2):159–64. <https://doi.org/10.1038/nbt.2106> <http://www.nature.com/nbt/journal/v30/n2/abs/nbt.2106.html#supplementary-information>. PMID: 22252508
86. Cukuroglu E, Engin HB, Gursoy A, Keskin O. Hot spots in protein–protein interfaces: Towards drug discovery. *Progress in Biophysics and Molecular Biology*. 2014; 116(2–3):165–73. <https://doi.org/10.1016/j.pbiomolbio.2014.06.003> PMID: 24997383
87. Buyong M, Ruth N. Druggable Orthosteric and Allosteric Hot Spots to Target Protein-protein Interactions. *Current Pharmaceutical Design*. 2014; 20(8):1293–301. <http://dx.doi.org/10.2174/13816128113199990073>. PMID: 23713780
88. Thangudu RR, Bryant SH, Panchenko AR, Madej T. Modulating Protein–Protein Interactions with Small Molecules: The Importance of Binding Hotspots. *Journal of Molecular Biology*. 2012; 415(2):443–53. <https://doi.org/10.1016/j.jmb.2011.12.026> PMID: 22198293
89. Arkin Michelle R, Tang Y, Wells James A. Small-Molecule Inhibitors of Protein-Protein Interactions: Progressing toward the Reality. *Chemistry & Biology*. 2014; 21(9):1102–14. <https://doi.org/10.1016/j.chembiol.2014.09.001>.
90. Pellegrino M, Mancini F, Lucà R, Coletti A, Giacchè N, Manni I, et al. Targeting the MDM2/MDM4 Interaction Interface as a Promising Approach for p53 Reactivation Therapy. *Cancer Research*. 2015; 75(21):4560–72. <https://doi.org/10.1158/0008-5472.CAN-15-0439> PMID: 26359458
91. Wu W, Xu C, Ling X, Fan C, Buckley BP, Chernov MV, et al. Targeting RING domains of Mdm2-MdmX E3 complex activates apoptotic arm of the p53 pathway in leukemia/lymphoma cells. *Cell Death Dis*. 2015; 6:e2035. <https://doi.org/10.1038/cddis.2015.358> PMID: 26720344

92. Emami KH, Nguyen C, Ma H, Kim DH, Jeong KW, Eguchi M, et al. A small molecule inhibitor of β -catenin/cyclic AMP response element-binding protein transcription. *Proceedings of the National Academy of Sciences of the United States of America*. 2004; 101(34):12682. <https://doi.org/10.1073/pnas.0404875101> PMID: 15314234
93. Canner JA, Sobo M, Ball S, Hutzen B, DeAngelis S, Willis W, et al. MI-63: A novel small-molecule inhibitor targets MDM2 and induces apoptosis in embryonal and alveolar rhabdomyosarcoma cells with wild-type p53. *British Journal Of Cancer*. 2009; 101:774. <https://doi.org/10.1038/sj.bjc.6605199> PMID: 19707204
94. Tse C, Shoemaker AR, Adickes J, Anderson MG, Chen J, Jin S, et al. ABT-263: A Potent and Orally Bioavailable Bcl-2 Family Inhibitor. *Cancer Research*. 2008; 68(9):3421. <https://doi.org/10.1158/0008-5472.CAN-07-5836> PMID: 18451170