Reliability Engineering and System Safety 200 (2020) 106899

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

# Efficient probabilistic multi-objective optimization of complex systems using matrix-based Bayesian network

Ji-Eun Byun[a], Junho Song[b,*]

[a] Department of Civil, Environmental and Geomatic Engineering, University College London, London, UK
[b] Department of Civil and Environmental Engineering, Seoul National University, Seoul, South Korea

ARTICLE INFO

*Keywords:*
Approximate optimization
Complex systems
Influence diagram
Matrix-based Bayesian network (MBN)
Multi-objective decision-making
System optimization

ABSTRACT

For optimal design and maintenance of complex systems such as civil infrastructure systems or networks, the optimization problem should take into account the system-level performance, multiple objectives, and the uncertainties in various factors such as external hazards and system properties. Influence Diagram (ID), a graphical probabilistic model for decision-making, can facilitate modeling and inference of such complex problems. The optimal decision rule for ID is defined as the probability distributions of decision variables that minimize (or maximize) the sum of the expected values of utility variables. However, in a discrete ID, the interdependency between component events that arises from the definition of the system event, results in the exponential order of complexity in both quantifying and optimizing ID as the number of components increases. In order to address this issue, this paper employs the recently proposed matrix-based Bayesian network (MBN) to quantify ID for large-scale complex systems. To reduce the complexity of optimization to polynomial order, a proxy measure is also introduced for the expected values of utilities. The mathematical condition that makes the optimization problems employing proxy objective functions equivalent to the exact ones is derived so as to promote its applications to a wide class of problems. Moreover, the proposed proxy measure allows the analytical evaluation of a set of non-dominated solutions in which the weighted sum of multiple objective values is optimized. By using the strategies developed to compensate the errors by the approximation as well as the weighted sum formulation, the proposed methodology can identify even a larger set of non-dominated solutions than the exact objective function of weighted sum. Four numerical examples demonstrate the accuracy and efficiency of the proposed methodology. The supporting source code and data are available for download at https://github.com/jieunbyun/GitHub-MBN-DM-code.

## 1. Introduction

In various efforts to construct, operate, and maintain real-world complex systems such as civil infrastructures and their networks, it is crucial to identify optimal decision-making strategies especially when budgets are limited. When dealing with such systems, the optimization should be able to take into account not only the performance of individual components but also the system-level performance. However, mathematical formulations for the system-level optimization are not straightforward in general. In addition, for large-scale systems, multiple random variables (r.v.'s) are introduced to represent both external factors (e.g. natural or man-made hazards and deterioration-inducing environment) and internal factors (e.g. material or geometric properties of components and system), for which a high-dimensional joint probability distribution needs to be constructed. Probabilistic graphical models (PGMs) can be employed to this end, which can translate real-world causal relationships into mathematical representations [1,2].

Bayesian network (BN) is one of the most widely used PGMs, which facilitates the modeling of causal relationships between r.v.'s as a joint probability distribution by use of nodes and directed arrows (Fig. 1(a)). On the other hand, Influence Diagram (ID) is an extension of BN for the purpose of decision-making in which two additional types of variables are introduced, namely, decision and utility variables (Fig. 1(b)). As the terms imply, decision variables represent the decision alternatives while utility variables quantify the utilities of each instance of interest. The optimal decision rule for an ID is defined as the probability mass functions (PMFs) of decision variables that minimize (or maximize) the sum of the expectations of utility variables. Discrete BN and ID in which all r.v.'s are discrete, allow us to develop inference algorithms that are flexible and widely applicable.

* Corresponding author.
*E-mail address:* junhosong@snu.ac.kr (J. Song).

https://doi.org/10.1016/j.ress.2020.106899
Received 16 April 2019; Received in revised form 19 February 2020; Accepted 22 February 2020
Available online 24 February 2020

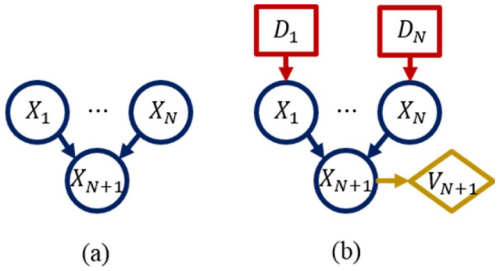


**Fig. 1.** Converging structure of (a) BN and (b) example ID for optimizing system-level performance.

BN and ID representing complex systems are often characterized by converging structures between the r.v.'s standing for component events, $\boldsymbol{X}_N = \{X_1, ..., X_N\}$, and that for system event, $X_{N+1}$, as illustrated in Fig. 1. This is because the system state is determined by the combinatorial states of components. In discrete BN and ID, such structure requires quantifying the conditional PMF of system event given each joint state of components, i.e. $P(X_{N+1}|\boldsymbol{X}_N)$. Therefore, an exponential number of parameters are required to this end as the number of components increases. Furthermore, this converging structure results in the exponential order of complexity for optimization of ID as the number of components increases. As illustrated in Fig. 1(b), when the system performance is considered as the utility variable representing a decision criterion ($V_{N+1}$), the decision alternatives for each component ($D_1$, ..., $D_N$) cannot be optimized separately, i.e. local optimality for each decision variable does not guarantee the global optimality. Therefore, the optimization needs to search all joint assignments of the decision variables.

For efficient quantification of $P(X_{N+1}|\boldsymbol{X}_N)$, several methodologies have been developed to exploit the regularity in the definition of system event. For instance, Tien and Der Kiureghian [3] employed data compression techniques for efficient storage of PMFs, and Bensi and Der Kiureghian [2] modified the graphical structure of BN using cut- and link-sets for network connectivity problem so that the number of arrows converging to the system-event can be reduced. On the other hand, Zhang and Pool [4] proposed to quantify PMFs by factorizing each instance of system event to avoid naïve enumeration of all instances, which is often redundant especially for large-scale systems. Recently, in order to unify and generalize these previous efforts and to enable efficient implementation of discrete BN, the matrix-based BN (MBN) has been developed [5]. MBN stores PMFs in the form of matrices to facilitate a straightforward modeling of the regularity in a system event. Moreover, both exact and approximate inference algorithms were developed for consistent applications. In this paper, the MBN is employed to quantify and inference large-scale systems.

On the other hand, for efficient optimization of ID, several algorithms have been proposed and expanded for larger systems [6-8]. However, the applicable size of problems is still significantly limited, and thus such approaches are unable to deal with large-scale complex systems including infrastructures and structural systems. This issue has prevented us from performing the optimization with direct consideration of the system-level performance, and thus previous efforts have been limited to either sequential optimization [9] or approximate system analysis [10]. In order to address this issue, a proxy measure for the expectation of a utility variable is proposed in this paper, by which the optimization problem can be decomposed into individual decision variables, i.e. approximately achieving the global optimality through local optimality. It is mathematically shown that the optimization problems using exact and proxy objective functions should be equivalent to each other in most problem settings. In addition, heuristic strategies can be developed to numerically check such equivalence in a given problem and to compensate the approximation errors.

Finally, decision-making processes for complex systems often consider more than one objective, e.g. cost and system performance. These conflicting objectives can be handled by having the objective function represent one of the objectives while the others are considered as constraints [11]. While this strategy identifies a single optimal solution, in practice, a set of non-dominated solutions – also called Pareto solutions or Pareto surface – might be of interest as it is often not straightforward to determine a pre-specified level or bounds for constraints in advance, e.g. budget limit or the upper bound of system failure probability. Diehl and Haimes [12] developed an algorithm to identify all non-dominated solutions for discrete ID, but the algorithm is limited to the applications of relatively small size of problems. Heuristic algorithms such as genetic algorithm (GA) can also be useful to this end [13]. However, they might be inefficient especially when the computational cost to evaluate objective values is high, and it is often not straightforward to conclude the convergence of solutions. The proxy measure proposed in this paper allows the analytical evaluation of a set of non-dominated solutions by optimizing the weighed sum of objective values. Moreover, by developing an iterative optimization scheme, the approximation errors and the possibility of missing non-dominated solutions by weighted sum formulation, can be compensated to obtain more accurate and diverse solutions.

The rest of the paper is organized as follows. Section 2 provides a brief introduction of the BN, ID, and MBN methodologies in general, and discusses the central issues addressed in this paper. The extension of the MBN for ID is also provided, aiming for efficient quantification of ID. Next, the proxy objective function is proposed for the efficient optimization of ID in Section 3, along with the mathematical derivations to assess the performance of the proposed alternative function. Section 4 illustrates the multi-objective optimization scheme proposed to obtain a set of non-dominated solutions by utilizing the proposed proxy function. This section also provides the performance analysis of the scheme as well as heuristic strategies to improve the performance. In Section 5, four numerical examples illustrate and demonstrate the proposed optimization methodology. Two examples of relatively small sizes demonstrate the procedure and accuracy of the proposed methodology while the other two examples test the efficiency as well, through the comparison with GA. The examples also demonstrate the applicability of the MBN to large-scale civil systems with complex definitions of system events. Finally, Section 6 summarizes the paper and provides concluding remarks. In the formulations of this paper, the upper and lower cases respectively denote the r.v.'s and their assignments, e.g. $X$ and $x$. The letters are bolded when a set of r.v.'s are referred to. For simplicity, the assignment of specific value over a r.v. is denoted by a superscript, e.g. $X = 0 \Leftrightarrow x^0$.

## 2. Quantification of influence diagram (ID) for complex systems using matrix-based Bayesian network (MBN)

### 2.1. Background: Bayesian network (BN) and ID for complex systems

In a BN, circular nodes stand for the r.v.'s while their causal relationships are represented by arrows directed from the causal r.v.'s (parent nodes) to resultant ones (child nodes) [1]. In the following discussions, the terms *r.v.* and *node* are used synonymously. A discrete BN over a set of r.v.'s, $\boldsymbol{X}$, quantifies each r.v. $X \in \boldsymbol{X}$ by a PMF conditioned on the set of its parent nodes, $Pa_X \subset \boldsymbol{X}$. The joint PMF represented by BN is the product of such conditional PMFs of each r.v. $X \in \boldsymbol{X}$, i.e.

$$P(\boldsymbol{X}) = \prod_{X \in \boldsymbol{X}} P(X|Pa_X) \tag{1}$$

while $P(X|Pa_X)$ is the marginal PMF of $X$ when $Pa_X = \emptyset$.

ID introduces two types of variables in addition to r.v.'s, namely, *utility* and *decision* variables. Utility variables $V \in \boldsymbol{V}$, are deterministic functions of the assignments over their parent nodes, and represented by rhombuses in the graph. In this paper, the symbols to represent the utility variables are also used to refer to the corresponding

deterministic functions, i.e. $V(Pa_V)$. On the other hand, the decision variables $D \in \boldsymbol{D}$, stand for the decision alternatives of consideration, which are illustrated by squares in the graph. A *decision rule* for $D$ refers to the conditional PMF $P(D|Pa_D)$ which is determined in a way to minimize (or maximize) the sum of the expectations of utility variables.

### 2.2. Problem definition

The proposed optimization method and the following derivations assume three properties of a given discrete ID. First, decision rules are assumed to have no parent nodes, which is the case for most practical applications. Secondly, the decision variables do not share their child nodes; if they do, they can be regarded as a single variable during optimization. Finally, the rules are considered *deterministic*, i.e. $P(D)$ assigns nonzero probability to exactly one value of $D$. Accordingly, a decision rule is equivalent to an assignment $\boldsymbol{d}$ over $\boldsymbol{D}$, and the optimization for ID is formulated as

$$\min_{\boldsymbol{d} \in Val(\boldsymbol{D})} \sum_{V \in \boldsymbol{V}} E[V|\boldsymbol{d}] \tag{2}$$

where $E[V|\boldsymbol{d}]$ is the expectation of $V$ given $\boldsymbol{d}$, and $Val(\cdot)$ is the set of values that the corresponding variable can take.

In spite of the general applicability of the proposed method, this paper illustrates the formulations and applications with a focus on system events that can be characterized by the ID in Fig. 2, which assumes two main features. First, two types of utility variables are considered, namely, cost $V_n$ for $D_n$, $n = 1, \ldots, N$, and $V_{N+1}$ that quantifies the system failure probability, i.e.

$$V_{N+1}(X_{N+1}) = \begin{cases} 1, & X_{N+1} = 0 \\ 0, & otherwise \end{cases} \tag{3}$$

where the r.v.'s $X_n$, $n = 1, \ldots, N$, and $X_{N+1}$ respectively stand for the events of the $n$th component and the system. Secondly, a decision variable $D_n$ is connected to only one r.v. $X_n$, which in practical sense, implies that decision alternatives are separately considered for each component. Additional r.v.'s can be added to this basic model to account for other factors, e.g. hazards, external loads, and material properties, as illustrated in the numerical examples. Following the definitions of utilities introduced in this paper, minimization problem is considered as formulated in Eq. (2) while maximization problem is also common for ID.

#### 2.2.1. Issues caused by converging structure

The state of a system is determined by the joint states of its components, which was explained by the converging structure in BN and ID between nodes for the components and the system, as illustrated in Fig. 1. Since complex systems usually consist of a large number of components, such converging structure is the main source of difficulties in both the quantification of BN and the optimization of ID. The quantification of conditional PMF $P(X_{N+1}|\boldsymbol{X}_N)$, requires the probability values for each joint state of the r.v.'s $\boldsymbol{X}_N \cup \{X_{N+1}\}$, whose number increases exponentially as $N$ increases. This issue can be addressed by exploiting the regularity in system event to achieve an efficient quantification. In addition, when the complete quantification is still infeasible due to the large number of components and complex definition of system event, approximate inference can be performed in which only a subset of instances is utilized. The MBN can handle these tasks as illustrated in Section 2.2 of [5].

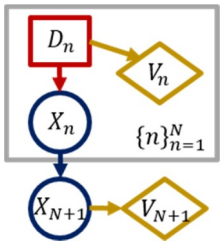


**Fig. 2.** Basic structure of ID for complex systems.

On the other hand, a decision variable $D_{n_1} \in \boldsymbol{D}$ is said to be *strategically relevant* to another decision variable $D_{n_2} \in \boldsymbol{D}$ if a decision rule $d_{n_1}$ is optimal for the decision rule $(\boldsymbol{d}_{-n}, d_{n_2})$, but not for $(\boldsymbol{d}_{-n}, d'_{n_2})$ for two different assignments $d_{n_2}, d'_{n_2} \in Val(D_{n_2})$ where $\boldsymbol{D}_{-n} = \boldsymbol{D} - \{D_{n_1}, D_{n_2}\}$, i.e. the local optimality of $d_{n_1}$ does not guarantee its global optimality [1]. In Fig. 2, the utility variables $V_n$, $n = 1, \ldots, N$, do not make the decision variables to be strategically relevant as the value of $V_n$ is determined only by $D_n$. However, in optimizing the expectation of $V_{N+1}$, the $N$ decision variables become strategically relevant to each other through the converging structure. This raises the need for examining all joint assignments over $\boldsymbol{D} = \{D_1, \ldots, D_N\}$ whose number again exponentially increases as $N$ increases. In order to address this issue, a proxy measure for $E[V_{N+1}|\boldsymbol{d}]$ is proposed in Section 3 for approximate but efficient optimization.

### 2.3. Extension of MBN to ID

The MBN has been recently proposed as an alternative data structure of conditional PMFs in discrete BN for efficient quantification and unification of inference methodologies, especially exact and approximate inferences [5]. The MBN quantifies PMFs using conditional probability matrices (CPMs) $\mathcal{M} = \langle C; \mathbf{p} \rangle$ which consist of event matrix $\mathbf{C}$ and probability vector $\mathbf{p}$. Each instance of interest is considered a rule $\rho = \langle \boldsymbol{c}; p \rangle$ in the MBN, where $\boldsymbol{c}$ and $p$ are respectively the corresponding assignment and probability, and they are jointly stored in the same rows of $\mathbf{C}$ and $\mathbf{p}$. While the data format of matrix facilitates the practical implementation, $\mathcal{M}$, $\mathbf{C}$, and $\mathbf{p}$ can be conceptually regarded as the set of rules ρ, assignments $\boldsymbol{c}$, and probabilities $p$ of instances of interest, respectively.

There are mainly two distinctive features of MBN. First, MBN introduces "−1" *state* to exploit the context-specific independence, i.e. originally dependent variables become independent when some specific values are assigned to a subset of variables. For example, consider the example system with three components in Fig. 3, whose states are represented by the r.v.'s $\boldsymbol{X}_N$ where $N = 3$, with the value of 1 for survival and 0 for failure. The system event is defined as the connectivity between the two nodes $s$ and $t$, and the r.v. for system event, $X_{N+1}$ takes the value of 1 for being connected, and 0 otherwise. Then, the CPM $\mathcal{M}_{X_{N+1}} = \langle \mathbf{C}_{X_{N+1}}; \mathbf{p}_{X_{N+1}} \rangle$ for $X_{N+1}$ is quantified as

$$\boldsymbol{C}_{X_{N+1}} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix} \text{ and } \boldsymbol{p}_{X_{N+1}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{4}$$

where a r.v. $X$ (a set of r.v.'s, $\boldsymbol{X}$) in the subscript of a CPM implies that the CPM quantifies the conditional PMF $P(X|Pa_X)$, i.e. $P(\boldsymbol{X}|Pa_{\boldsymbol{X}})$. The columns of $\mathbf{C}_{X_{N+1}}$ sequentially represent the states of $X_{N+1}$, $X_1$, $X_2$, and $X_3$. For example, the first row of $\mathbf{C}_{X_{N+1}}$ states that the two nodes $s$ and $t$ are connected with each other (i.e. $x^1_{N+1}$) as long as the first component survives ($x^1_1$) regardless of the states of $X_2$ and $X_3$. Since the system state is a deterministic function of component states in this example, the probability values in $\mathbf{p}_{X_{N+1}}$ are all one.

It should be noted that the instances in a CPM have to be mutually exclusive (the quantification scheme to ensure this condition is discussed in [5]). Accordingly, in the following three instances in Eq. (4),

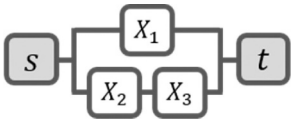


**Fig. 3.** Example system with 3 components.

$X_1$ is assigned 0 so as to be disjoint with the rule represented by the first row. This example demonstrates that by exploiting context-specific independence using "−1" state, there is no need for explicit enumeration of all instances. That is, 4 instances are sufficient to quantify a total of $|Val(\boldsymbol{X}_N)|=2^3$ instances where $|\boldsymbol{X}|$ denotes the cardinality of set $\boldsymbol{X}$. The basic BN operations, i.e. compatibility, conditioning, sum, and product of PMFs, have been modified to be compatible with "−1" state of MBNs [5].

Another important characteristic of the MBN is that it does not require the CPMs to include the exhaustive set of instances, but only to include the instances of interest, i.e. the ones that affect the inference results. The instances that are not included in CPMs, are considered to have zero probabilities during the inference by MBN. For instance, in Eq. (4), the rule $< (x_{N+1}^0, x_1^1, x_2^{-1}, x_3^{-1}); 0>$ is not stored as the instance always has zero probability and thereby, has no effect on the inference result. As demonstrated by this example, this relaxation is useful particularly for deterministic functions. Furthermore, in the optimization of ID, the CPMs exclude the instances that do not affect the optimization results. For example, by the definition of $V_{N+1}(X_{N+1})$ in Eq. (3), the first and fourth rules in $\mathcal{M}_{X_{N+1}}$ do not affect the optimization results, which further reduces the CPM $\mathcal{M}_{X_{N+1}}$ to

$$\mathbf{C}_{X_{N+1}} = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{and } \mathbf{p}_{X_{N+1}} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{5}$$

This relaxation also allows the application of CPMs using approximate inference, which removes the boundary between the implementations of exact and approximate inferences as demonstrated by the numerical example in Section 5.4.

The extension of MBN to ID is straightforward as the utility variables are deterministic functions of their parent nodes while the PMFs for decision variables do not have to be quantified in advance, but are optimized afterwards. For instance, from the definition of $V_{N+1}$ in Eq. (3), the CPM $\mathcal{M}_{V_{N+1}} = <\mathbf{C}_{V_{N+1}}; \mathbf{p}_{V_{N+1}}>$ is quantified as

$$\mathbf{C}_{V_{N+1}} = [1 \;\; 0] \text{and} \;\; \mathbf{p}_{V_{N+1}} = [1] \tag{6}$$

where the first and second columns of $\mathbf{C}_{V_{N+1}}$ refer to the assignments over $V_{N+1}$ and $X_{N+1}$, respectively. It is noted that $\mathcal{M}_{V_{N+1}}$ does not include the rule for $V_{N+1}(x_{N+1}^1) = 0$ as it is not utilized during the optimization in analogy to the formulation of Eq. (5).

## 3. Proxy objective function for optimizing ID that has multiple strategically relevant decision variables

Recall that the decision variables become strategically relevant to optimize the expectations in Eq. (2). To eliminate such relevance, we propose to approximate the expectations to decompose the evaluation into each decision variable, i.e.

$$E[V|\boldsymbol{d}] \approx \tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}] = \sum_{n=1}^{N} E[V|d_n, \bar{\boldsymbol{d}}_{-n}] \tag{7}$$

where $\bar{\boldsymbol{d}}$ is a *basis decision rule*; and $\boldsymbol{d}_{-n}$ denotes the decision rule that is identical to $\boldsymbol{d}$, but not defined on $D_n$. In other words, the proxy measure conditions the expectation on $\bar{\boldsymbol{d}}$ and changes one decision variable at a time; then, it approximates the exact function by summing the $N$ expectations. As expected, the approximation accuracy is affected by the choice on $\bar{\boldsymbol{d}}$, for which the heuristic strategies are suggested in Section 4.2.

This proxy measure allows the optimization problem to be decomposed in terms of each decision variable for the operations of minimization/maximization and summation are exchangeable, i.e.

$$\begin{aligned} \min_{\boldsymbol{d}\in Val(\boldsymbol{D})} \sum_{V\in\boldsymbol{V}} \tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}] &= \min_{\boldsymbol{d}\in Val(\boldsymbol{D})} \sum_{V\in\boldsymbol{V}} \sum_{n=1}^{N} E[V|d_n, \bar{\boldsymbol{d}}_{-n}] \\ &= \sum_{n=1}^{N} \min_{d_n\in Val(D_n)} \sum_{V\in\boldsymbol{V}} E[V|d_n, \bar{\boldsymbol{d}}_{-n}] \end{aligned} \tag{8}$$

In other words, regarding the proxy objective function in Eq. (8), the local and the global optimality are equivalent, and the optimization can be simply done by finding the assignment $d_n \in Val(D_n)$ that minimizes $E[V|d_n, \bar{\boldsymbol{d}}_{-n}]$ for each decision variable $D_n$. As a result, the computational complexity can be reduced from the exponential order to the linear one with respect to $N$, i.e. from $O(\prod_{n=1}^{N}|Val(D_n)|)$ to $O(\sum_{n=1}^{N}|Val(D_n)|)$.

The proposed method is analogous to the coordinate descent (CG) algorithm in that it attempts to find the optimal solution by changing one variable at a time with other variables fixed. However, while the CG algorithm sequentially moves the fixed point after identifying the optimal value of a decision variable, the proposed method does not move the point incrementally but rather optimizes the $N$ functions all together. As illustrated later in this paper, the approximation error can be compensated by iterating multiple basis decision rules and the proposed heuristic strategies.

### 3.1. Performance analysis

For the two optimization problems respectively defined in terms of $E[V|\boldsymbol{d}]$ in Eq. (2) and $\tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]$ in Eq. (7) to be equivalent, the proxy objective functions should have a positive sensitivity in regard to the exact function, i.e.

$$\text{If } \; E[V|\boldsymbol{d}_1] \geq E[V|\boldsymbol{d}_2], \;\; \tilde{E}[V|\boldsymbol{d}_1;\bar{\boldsymbol{d}}] \geq \tilde{E}[V|\boldsymbol{d}_2;\bar{\boldsymbol{d}}] \;\; \text{for} \;\; \forall\, \boldsymbol{d}_1, \boldsymbol{d}_2 \in Val(\boldsymbol{D}) \tag{9}$$

for some basis decision rule $\bar{\boldsymbol{d}} \in Val(\boldsymbol{D})$. The following illustrations demonstrate that the positive correlations are likely to take place in most settings by deriving the derivative of the proxy function with respect to the exact one; in other words, it is shown that the derivative is in general likely to be positive.

To this end, for a BN over r.v.'s $\boldsymbol{X}$, the evaluation of $E[V|\boldsymbol{d}]$ is expressed in terms of the assignments, instead of the r.v.'s as in Eq. (1), i.e.

$$E[V|\boldsymbol{d}] = \sum\nolimits_{\boldsymbol{c}\in\mathbf{C}_{\boldsymbol{X}}} V(\boldsymbol{c})\cdot P(\boldsymbol{c}|\boldsymbol{d}) \tag{10}$$

where $\boldsymbol{c}$ is a row of the event matrix; $V(\boldsymbol{c})$ is the abbreviation of $V(\boldsymbol{c}<Pa_V>)$; and $\boldsymbol{c}<\boldsymbol{X}>$ is the assigned values over $\boldsymbol{X}$ by $\boldsymbol{c}$. The probability term in the equation is evaluated as

$$P(\boldsymbol{c}|\boldsymbol{d}) = \prod_{X\in\boldsymbol{X}} P(\boldsymbol{c} < X > |\boldsymbol{c} < Pa_X > ,\boldsymbol{d}) \tag{11}$$

Although it is often infeasible to evaluate such joint distribution over all r.v.'s, this formulation is useful for the purpose of demonstration. That is, changing the assignment of decision variable $D_n$, $n = 1, \ldots, N$, can be regarded as changing the probability terms in Eq. (11). Accordingly, to proceed the demonstration, the terms in Eq. (11) should be decomposed as a function of each decision variable $D_n$ and its child nodes $Ch_n$, i.e.

$$\begin{aligned} P(\boldsymbol{c}|d_n, \boldsymbol{d}_{-n}) &= \prod_{X\in Ch_n} P(\boldsymbol{c} < X > |\boldsymbol{c} < Pa_X > ,d_n, \boldsymbol{d}_{-n})\cdot \\ &\quad \prod_{Y\in\boldsymbol{X}-Ch_n} P(\boldsymbol{c} < Y > |\boldsymbol{c} < Pa_Y > ,\boldsymbol{d}_{-n}) \\ &= P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n})\cdot P(\boldsymbol{c} < \boldsymbol{X} - Ch_n > |\boldsymbol{d}_{-n}) \end{aligned} \tag{12}$$

By considering those probabilities as continuous variables, Eq. (9) is equivalent to the condition

$$\frac{\partial \tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]}{\partial E[V|\boldsymbol{d}]} = \sum_{n=1}^{N}\sum_{\boldsymbol{c}\in\mathbf{C}_X}\left(\frac{\partial \tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]}{\partial P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})}\right) \Big/ \left(\frac{\partial E[V|\boldsymbol{d}]}{\partial P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n})}\right) \geq 0,\ \forall\ \boldsymbol{d} \in Val(\boldsymbol{D}) \tag{13}$$

Before deriving the partial derivatives, it should be taken into account that the degree of freedom in determining the values of $P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n})$, as well as $P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})$, is one less than $|Val(\boldsymbol{X})|$ for each $d_n$, from the axiom that the sum of probabilities must be one. To reflect such constraint, an assignment $\hat{\boldsymbol{c}} \in \mathbf{C}_{\boldsymbol{X}}$ without "−1" state (note that assignments with "−1" state stand for more than one instance) is selected, leading to the relationship with other instances $\boldsymbol{c} \in \mathbf{C}_{\boldsymbol{X}} - \{\hat{\boldsymbol{c}}\}$ as

$$P(\hat{\boldsymbol{c}} < Ch_n > |d_n, \boldsymbol{d}_{-n}) = (1-\delta) - \sum_{\boldsymbol{c}\in\mathbf{C}_{\boldsymbol{X}}-\{\hat{\boldsymbol{c}}\}} P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n}) \tag{14}$$

The constant $\delta \geq 0$ does not affect the results of differentiation, but is introduced to keep the equation valid not only when the CPM $\mathcal{M}_{\boldsymbol{X}}$ is exhaustive, i.e. $\sum_{\boldsymbol{c}\in\mathbf{C}_{\boldsymbol{X}}} P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n}) = 1$ and $\delta = 0$, but also when $\mathcal{M}_{\boldsymbol{X}}$ is not, i.e. $\sum_{\boldsymbol{c}\in\mathbf{C}_{\bar{\boldsymbol{X}}}} P(\boldsymbol{c} < Ch_n > |d_n, \boldsymbol{d}_{-n}) < 1$ and $\delta > 0$. Then, from Eqs. (10), (12), and (14), the partial derivative of $E[V|\boldsymbol{d}] = E[V|d_n, \boldsymbol{d}_{-n}]$ with regard to $P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})$ is derived for some decision rule $\bar{\boldsymbol{d}}$, $\forall\ \boldsymbol{c} \in \mathbf{C}_{\boldsymbol{X}} - \{\hat{\boldsymbol{c}}\}$, $\forall d_n \in Val(D_n)$, and $\forall\ n = \{1,...,N\}$ as

$$\begin{aligned}\frac{\partial E[V|\boldsymbol{d}]}{\partial P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})} &= \frac{\partial E[V|\boldsymbol{d}]}{\partial P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})} + \frac{\partial E[V|\boldsymbol{d}]}{\partial P(\hat{\boldsymbol{c}} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})} \cdot \frac{\partial P(\hat{\boldsymbol{c}} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})}{\partial P(\boldsymbol{c} < Ch_n > |d_n, \bar{\boldsymbol{d}}_{-n})} \\ &= E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,d_n, \bar{\boldsymbol{d}}_{-n}] - E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,d_n, \bar{\boldsymbol{d}}_{-n}] \\ &= E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\bar{\boldsymbol{d}}_{-n}] - E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\bar{\boldsymbol{d}}_{-n}]\end{aligned} \tag{15}$$

In the second line of the equation, the fact is utilized such that differentiating a joint distribution by the probability of a certain assignment has the effect of conditioning the joint distribution on the assignment [14]. The third line reflects that the expectations become independent of $D_n$ given that the child nodes are specified by a certain value (recall that the decision variables do not share their child nodes). Using the partial derivative, Eq. (13) is derived as

$$\frac{\partial \tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]}{\partial E[V|\boldsymbol{d}]} = \sum_{n=1}^{N}\sum_{\boldsymbol{c}\in\mathbf{C}_X} \frac{E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\bar{\boldsymbol{d}}] - E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\bar{\boldsymbol{d}}]}{E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}] - E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\boldsymbol{d}]},\ \forall\ \boldsymbol{d} \in Val(\boldsymbol{D}) \tag{16}$$

For most settings of $\boldsymbol{c} \in \mathbf{C}_X - \{\hat{\boldsymbol{c}}\}$ and $\boldsymbol{d} \in Val(\boldsymbol{D})$, the denominators and numerators inside the summation are likely to have the same signs, i.e. the fractions are likely to be positive. First, when $\boldsymbol{c} < Pa_V > = \hat{\boldsymbol{c}} < Pa_V >$, i.e. $V(\boldsymbol{c}) = V(\hat{\boldsymbol{c}})$, it is likely that the change in the assignment over certain variables $Ch_n$ leads to either a decrease or increase of the expected values of $V$ regardless of the chosen decision rules. For example, in the ID in Fig. 2, consider that $V_{N+1}$ quantifies the system failure probability. Accordingly, the change from $x_n^0$ to $x_n^1$ for some $X_n \in \boldsymbol{X}_{\boldsymbol{N}}$, is likely to decrease the expected values of $V_{N+1}$ given any decision rules. On the other hand, when $\boldsymbol{c} < Pa_V > \neq \hat{\boldsymbol{c}} < Pa_V >$ and $V(\boldsymbol{c}) \neq V(\hat{\boldsymbol{c}})$, regardless of the assignment $\boldsymbol{c} < Ch_n >$ and decision rule $\boldsymbol{d}$, it is likely that $E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}] > E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\boldsymbol{d}]$ for $V(\boldsymbol{c}) > V(\hat{\boldsymbol{c}})$, and $[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}] < E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\boldsymbol{d}]$ for $V(\boldsymbol{c}) < V(\hat{\boldsymbol{c}})$. This is especially the case when the difference between utility values is more significant than that between probability values. For instance, since $V(x_{N+1}^0) - V(x_{N+1}^1) = 1$ with $V(x_{N+1}^1) = 0$, the inequality $E[V(x_{N+1}^0)|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}] > E[V(x_{N+1}^1)|\hat{\boldsymbol{c}} < Ch_n > ,\boldsymbol{d}]$ holds for all $\boldsymbol{c} \in \mathbf{C}_{\boldsymbol{X}} - \{\hat{\boldsymbol{c}}\}$ and $\boldsymbol{d} \in Val(\boldsymbol{D})$ as long as $P(x_{N+1}^0|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}) > 0$.

A potentially more problematic issue than the signs of the fractions, is the presence of singular points, i.e. $E[V(\boldsymbol{c})|\boldsymbol{c} < Ch_n > ,\boldsymbol{d}] - E[V(\hat{\boldsymbol{c}})|\hat{\boldsymbol{c}} < Ch_n > ,\boldsymbol{d}] = 0$. This can happen when there are some combinations of $\boldsymbol{c} < Ch_n >$, $\hat{\boldsymbol{c}} < Ch_n >$, and $\boldsymbol{d}$ that make the two expectations equal. Also, although the empirical analysis in the previous paragraph is practically reasonable, the only possible strict demonstration is to inspect all possible configurations. Therefore, in order to address the possibility that the condition in Eq. (14) is not satisfied in some parts of the solution space, several basis decision rules $\bar{\boldsymbol{d}}$ can be utilized, and the final optimal solution can be selected by comparing the exact objective functions evaluated for the obtained optimal solutions. Moreover, even when Eq. (14) is not strictly satisfied, it is still likely that large proxy measures generally imply large exact values. To exploit such general tendency, multiple suboptimal solutions can be evaluated for each $\bar{\boldsymbol{d}}$, e.g. up to the fifth optimal solutions, which can also be analytically evaluated with respect to Eq. (13). These solutions can be compared with each other to identify the optimal solution for the exact objective function. By plotting $\tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]$ and $E[V|\boldsymbol{d}]$ of those suboptimal solutions with $V$ for which approximation is introduced, as illustrated in the numerical examples of Sections 5.3 and 5.4, the condition of Eq. (9), i.e. the positive correlation between the two measures, can be numerically checked for a given problem, without increasing the computational cost.

## 4. Multi-objective optimization using proxy objective function

### 4.1. Optimization using weighted sum as objective function

To facilitate the illustration, the following discussions focus on the ID structure in Fig. 2; however, it is noted that the proposed framework applies to any ID that satisfies the three conditions in Section 2.2, including ID with more than two types of utility variables. The ID in Fig. 2 employs two types of utility variables: (1) cost of decisions $\sum_{V_n\in\{V_1,...,V_N\}} E[V_n|d_n]$, $n = 1, ...,N$, and (2) system failure probability $E[V_{N+1}|\boldsymbol{d}]$. These two different objectives can be simultaneously considered by optimizing their weighted sum, i.e.

$$\min_{\boldsymbol{d}\in Val(\boldsymbol{D})} \sum_{V_n\in\{V_1,...,V_N\}} E[V_n|d_n] + \lambda E[V_{N+1}|\boldsymbol{d}],\ \lambda > 0 \tag{17}$$

By varying the weight λ, a set of non-dominated solutions can be obtained. Since the decision variables are not strategically relevant to each other in regards to $\{V_1,...,V_N\}$, but they are in regards to $V_{N+1}$, the proxy objective function in Eq. (7) is introduced only for $E[V_{N+1}|\boldsymbol{d}]$. The optimization problem is then approximated to $N$ optimization problems each of which involves a single decision variables $d_n$, $n = 1, ...,N$, i.e.

$$\min_{d_n\in Val(D_n)} [E[V_n|d_n] + \lambda E[V|d_n, \bar{\boldsymbol{d}}_{-n}]],\ \lambda > 0 \tag{18}$$

By decomposing the problem in terms of individual decision variables, the optimal solution can be evaluated by comparing only $|Val(D_n)|$ number of decision alternatives separately for each $D_n \in \boldsymbol{D}$. Furthermore, this is equivalent to mapping those alternatives into the space whose axes respectively represent the objective functions in Eq. (18), i.e. $E[V|d_n, \bar{\boldsymbol{d}}_{-n}]$ and $E[V_n|d_n]$. Exploiting the fact that the weight λ is related to the slope of a line in such space, the approximate optimization in Eq. (18) enables us to efficiently evaluate the values of λ that change the preference between these alternatives, as illustrated in the followings.

Specifically, consider the example case in Fig. 4 where $Val(D_n) = \{d_n^1,...,d_n^6\}$. In the figure, the optimal solutions identifiable by Eq. (18) are marked by blue squares, while the non-optimal solutions according to Eq. (18) are denoted by orange triangles. (The optimal solutions that can be identified by the weighted-sum-based objective function are discussed in detail in Section 4.3.) In the figure, the values

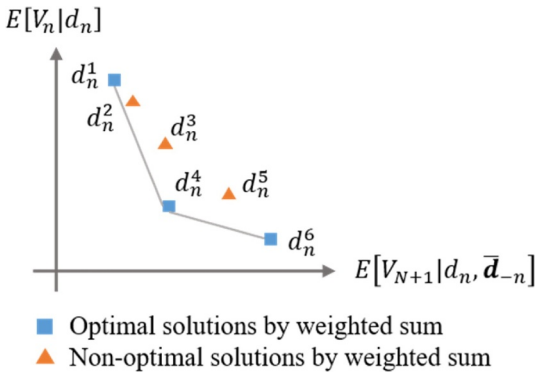


**Fig. 4.** Example space of $E[V_{N+1}|\boldsymbol{d}_n, \bar{\boldsymbol{d}}_{-n}]$ and $E[V_n|\boldsymbol{d}_n]$.

of λ at which the optimal solutions are changed, correspond to the absolute values of the negative slopes that connect the convex surface enveloping the solutions (grey lines). To evaluate such λ values, the solution with smallest $E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$, i.e. $d_n^1$ in Fig. 4, is set as the starting point as it is always one of the non-dominated solutions. Then, among the points with greater $E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$, the one having the most negative slope with respect to the current one, i.e. $d_n^4$ with respect to the current point $d_n^1$, is chosen as the next point. The identification is terminated at the point for which there are no remaining points with greater $E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$, or among them, no points with negative slopes as the search terminates at $d_n^6$ in the figure. By assembling the absolute values of such slopes for each $D_n$, all λ values that change the optimal solutions, can be identified.

It is noted that the order of computational cost required to identify a set of non-dominated solutions for the proxy function, is the same with that of evaluating the expectation $E[V_{N+1}|\boldsymbol{d}]$, i.e. the expectations $E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$ should be evaluated for $d_n \in Val(D_n)$ and $D_n \in \boldsymbol{D}$. It is noted that the evaluation can become efficient by designing the BN inference procedure to exploit the similar forms of the expectations. Moreover, the degree of accuracy is not affected by the number of utility variables for which the proxy function is adopted as long as the condition of Eq. (9) holds for the utilities, and the weights are separately introduced for each of them.

In analogy to the discussions for implementation in the last paragraph of Section 3.1, several $\bar{\boldsymbol{d}}$ can be utilized to compensate the errors by approximation, and an empirical strategy for selecting $\bar{\boldsymbol{d}}$ is proposed in Section 4.2. Moreover, one can get suboptimal solutions for each value of λ to compensate the errors and identify a larger set of Pareto solutions as discussed in Section 4.3. In the following numerical examples, up to the second suboptimal rules are evaluated. Finally, for a numerical check of the condition in Eq. (9), one can examine the positive correlation between $\tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]$ and $E[V|\boldsymbol{d}]$ of the obtained optimal solutions for the different values of λ. Since the quantities need to be evaluated in order to sort out the non-dominated solutions for the exact objective function, such examination does not increase the computational cost.

### 4.2. Selecting basis decision rules $\bar{\boldsymbol{d}}$

The proposed method can be considered as evaluating the weighted sum of arithmetic means instead of geometric means of the probability terms. The two means generally show positive correlation, but their difference can become larger, or even may show negative relationship as the spread of the terms becomes significant. For $\tilde{E}[V|\bar{\boldsymbol{d}}; \bar{\boldsymbol{d}}] = E[V|\bar{\boldsymbol{d}}]$, it can be stated that as a decision rule $\boldsymbol{d}$ becomes more different from $\bar{\boldsymbol{d}}$, it is more likely that $\tilde{E}[V|\boldsymbol{d};\bar{\boldsymbol{d}}]$ and $E[V|\boldsymbol{d}]$ have a weaker correlation and thus the condition of Eq. (9) may not hold.

Therefore, it is desirable to select the next $\bar{\boldsymbol{d}}$ that is most different from the current $\bar{\boldsymbol{d}}$ so that the solution space can be widely explored. To address this issue without complicating the implementation, we propose to select the one with the largest discrepancy among the non-dominated solutions obtained with $\bar{\boldsymbol{d}}$. After sorting the $N$ approximate optimal solutions $\boldsymbol{d}_k^*$, $k = 1, \ldots, N$, from the smallest to the largest values of $E[V|\boldsymbol{d}_k^*]$, the following is proposed as a measure of the discrepancy for a rule $\boldsymbol{d}_k^*$ from $\bar{\boldsymbol{d}}$:

$$D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}}) = -\frac{\tilde{E}[V|\boldsymbol{d}_{k+1}^*; \bar{\boldsymbol{d}}] - \tilde{E}[V|\boldsymbol{d}_k^*; \bar{\boldsymbol{d}}] + \epsilon}{E[V|\boldsymbol{d}_{k+1}^*] - E[V|\boldsymbol{d}_k^*] + \epsilon}, \quad \text{for } k = 1, \ldots, N-1 \tag{19}$$

where ϵ is a sufficiently small positive number to handle the case of $E[V|\boldsymbol{d}_{k+1}^*] = E[V|\boldsymbol{d}_k^*]$. In other words, the measure is equivalent to the negative of the sensitivity of the proxy measure with respect to the exact expectation. Accordingly, the one with the largest $D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}})$ is selected as next $\bar{\boldsymbol{d}}$. Since the terms in Eq. (19) are evaluated during the optimization, the procedure would demand a marginal increase in the computational cost.

However, when only the solutions identified as optimal are examined, there is a risk that a limited set of rules are repeatedly examined. In order to avoid such risk, it is proposed to use a rule alternately (1) that is randomly selected and (2) that maximizes the measure in Eq. (19) among the optimal solutions identified in the previous iteration.

### 4.3. Attainable solutions by proposed scheme

It is noted that, when the weighted sum is used as the objective function, some of the non-dominated solutions may not be identifiable. Specifically, in the space of objective values for a minimizing problem, consider the convex polyhedron $P = \{\boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) \in \mathbb{R}^p | \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) \geq b_i, \ i = 1, \ldots, q\}$ where $p$ is the number of objectives; $\boldsymbol{a}_i$, $i = 1, \ldots, q$, are the column vectors of dimension $p$; and $\boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}})$ is the vector of objective values in regards to some solution $\boldsymbol{d}$, i.e. $\boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) = (\tilde{E}[V_1|\boldsymbol{d};\bar{\boldsymbol{d}}], \ldots, \tilde{E}[V_p|\boldsymbol{d};\bar{\boldsymbol{d}}])$. In addition, suppose that $P$ includes all possible solutions – i.e. $\boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) \geq b_i$, $\forall i$, $\forall \boldsymbol{d}$, while its surface consists only of non-dominated solutions – i.e. for all $i$, the solution $\boldsymbol{d}$ that satisfies $\boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) = b_i$, must be a non-dominated solution. Then, the weighted sum formulation can identify only the solutions that are located on the surface of $P$, i.e. $\boldsymbol{d}$ that has some $i$ which satisfies $\boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x}(\boldsymbol{d}; \bar{\boldsymbol{d}}) = b_i$. For example, Fig. 5(a) illustrates the space of $E[V_{N+1}|\boldsymbol{d}]$ and $\sum_{n=1}^{N} E[V_n|d_n]$, and it is assumed that there are five non-dominated solutions $\boldsymbol{d}_k$, $k = 1, \ldots, 5$. The surface of such convex polyhedron enveloping and containing all solutions, are illustrated by solid grey lines. Since $\boldsymbol{d}_3$ is not on the surface, it cannot be identified as optimal by minimizing the weighted sum.

Interestingly, however, the re-mapping of solutions by the approximation in the proposed method makes it possible to identify a larger set of non-dominated solutions than the exact formulation. This is because the condition of Eq. (9) does not imply the conservation of relative gap between the values. Therefore, depending on the chosen basis decision rule $\bar{\boldsymbol{d}}$, the solutions can be re-mapped as illustrated in either Fig. 5(b) or (c). In both cases, the relative order of $E[V_{N+1}|\boldsymbol{d}]$ is not changed, i.e. Eq. (9) holds, but the sets of identifiable solutions are different. In other words, by utilizing multiple choices of $\bar{\boldsymbol{d}}$, the loss of solutions by weighted sum formulation can be compensated.

Moreover, as pointed out in the last paragraph of Section 4.1, by identifying several suboptimal solutions for each λ value, even a larger set of solutions can be identified. For example, the orange dotted line in Fig. 5(c) illustrates that by identifying the second optimal for the corresponding λ value, i.e. the absolute value of the slope, $\boldsymbol{d}_3$ can be identified as optimal. This is particularly advantageous when the Pareto surface for a given problem shows a highly nonlinear concave contour. This situation can occur when only a very small number of positive weights are obtained from the procedure discussed in Section 4.1.

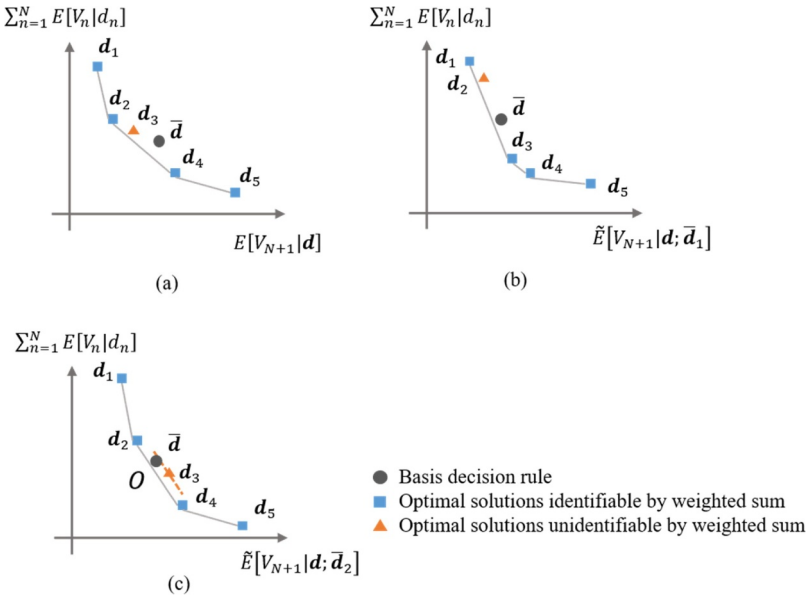


**Fig. 5.** Example space and solutions in regards to (a) $E[V_{N+1}|\boldsymbol{d}]$ and $\sum_{n=1}^{N} E[V_n|d_n]$, (b) $\tilde{E}[V_{N+1}|\boldsymbol{d};\bar{\boldsymbol{d}}_1]$ and $\sum_{n=1}^{N} E[V_n|d_n]$, and (c) $\tilde{E}[V_{N+1}|\boldsymbol{d};\bar{\boldsymbol{d}}_2]$ and $\sum_{n=1}^{N} E[V_n|d_n]$.

## 5. Numerical examples

In the following examples, up to the second optimal solutions are identified for each weight, except the illustrative example in Section 5.1 where only a single solution is examined. The strategy proposed in Section 4.2 is employed to select basis decision rules. In Sections 5.3 and 5.4, the results are compared with those by GA in which 50 populations are produced for each generation. Among them, 10 populations are randomly generated while two sets of 20 populations are generated respectively by cross-over and mutation of the non-dominated solutions in the previous population. For mutation, 15% of the assignments over decision variables $\boldsymbol{D}$ are randomly changed.

### 5.1. Illustrative example: a system with three components

Consider the system with three components in Fig. 3 whose ID is modeled as Fig. 6 with $N = 3$. The r.v. $H$ represents the intensity of the hazard (0 for insignificant and 1 for significant), and $D_n$ stands for the decision alternatives on the $n$th component (0 for doing nothing and 1 for retrofitting). The other variables are defined as illustrated in Section 2.3. The CPMs for $X_{N+1}$ and $V_{N+1}$ are defined respectively by Eqs. (5) and (6). On the other hand, the CPM $\mathcal{M}_H = <\mathbf{C}_H; \mathbf{p}_H>$ is set as

$$\mathbf{C}_H = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{and } \mathbf{p}_H = \begin{bmatrix} 0.8 \\ 0.2 \end{bmatrix} \tag{20}$$

and the CPMs $\mathcal{M}_{X_n} = <\mathbf{C}_{X_n}; \mathbf{p}_{X_n}>$ and $\mathcal{M}_{V_n} = <\mathbf{C}_{V_n}; \mathbf{p}_{V_n}>$ $n = 1, 2, 3$, are respectively given as

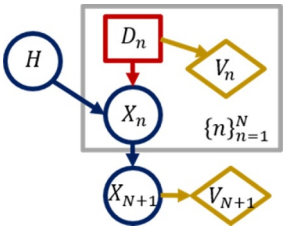


**Fig. 6.** ID for example system in Fig. 3.

$$\mathbf{C}_{X_n} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \text{and } \mathbf{p}_{X_n} = \begin{bmatrix} 0.2 \\ 0.8 \\ 0.1 \\ 0.9 \\ 0.4 \\ 0.6 \\ 0.3 \\ 0.7 \end{bmatrix}, \; n = 1, 2, 3 \tag{21}$$

where the first, second, and third columns of $\mathbf{C}_{X_n}$ represent the assignments over $X_n$, $D_n$, and $H$, respectively, and

$$\mathbf{C}_{V_1} = \begin{bmatrix} 0 & 0 \\ 100 & 1 \end{bmatrix}, \mathbf{C}_{V_2} = \begin{bmatrix} 0 & 0 \\ 60 & 1 \end{bmatrix}, \mathbf{C}_{V_3} = \begin{bmatrix} 0 & 0 \\ 50 & 1 \end{bmatrix}, \text{and } \mathbf{p}_{V_n} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \; n = 1, 2, 3 \tag{22}$$

where the first and second columns of $\mathbf{C}_{V_n}$ respectively stand for the assignments over $V_n$ and $D_n$.

When the basis rule is selected as $\bar{\boldsymbol{d}} = (\bar{d}_1, \bar{d}_2, \bar{d}_3) = (0, 0, 0)$, the expectations $E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$ for $d_n \in Val(D_n)$ and $n = 1, ..., N$ are evaluated as summarized in Table 1. Those expectations can be utilized to evaluate the values of $\lambda$ that change the optimal solutions together with the other type of utility variables, $V_n$, $n = 1, 2, 3$, e.g. for $D_1$,

$$\lambda = -\frac{100 - 0}{0.0672 - 0.1088} = 2,400 \tag{23}$$

In Eq. (23), such critical $\lambda$ value is simply the slope between the two solutions $d_1^0$ and $d_1^1$ for they are the only candidates. By evaluating the $\lambda$ values for other components as well, the optimal decision rules $\boldsymbol{d}_k^*$, $k = 1, ..., 4$, are obtained as

**Table 1**
$E[V_{N+1}|d_n, \bar{\boldsymbol{d}}_{-n}]$ for $d_n \in Val(D_n)$ and $n = 1, \cdots, N$.

| $n$ | 1 | 2 | 3 |
|---|---|---|---|
| $d_n^0$ | 0.1088 | 0.1088 | 0.1088 |
| $d_n^1$ | 0.0672 | 0.0912 | 0.0912 |

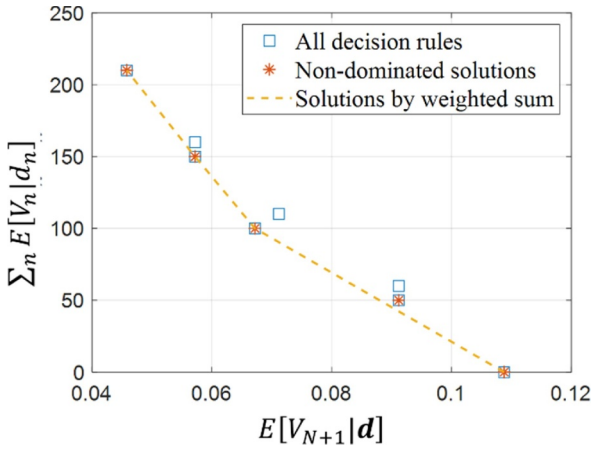


**Fig. 7.** All and optimal decision rules for example system in Fig. 3.

$$\boldsymbol{d}_1^* = (1, 1, 1) \text{ for } \lambda > 3,410; \quad \boldsymbol{d}_2^* = (1, 0, 1) \text{ for } 2,840 < \lambda < 3,410;$$
$$\boldsymbol{d}_3^* = (1, 0, 0) \text{ for } 2,400 < \lambda < 2,840; \text{ and } \boldsymbol{d}_4^* = (0, 0, 0) \text{ for } 0 < \lambda < 2,400 \tag{24}$$

There are $|Val(\boldsymbol{D})| = 8$ decision rules, and their exact values of $E[V_{N+1}|\boldsymbol{d}]$ and $\sum_{n=1}^{N} E[V_n|d_n]$ are illustrated in Fig. 7. Five non-dominated rules (red asterisks) are identified among the total of eight (blue squares). It is noted that the exact objective function of weighted sum (dotted yellow line) would identify four of the non-dominated rules. The rules obtained in Eq. (24) correspond to the four rules, and it should be noted that all solutions for the exact objective function can be identified by one iteration of the proposed method.

When the next iteration is desired, the new basis rule can be selected by evaluating the measure of discrepancy $D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}})$ in Eq. (19) for the solutions obtained by the approximate optimization. Specifically, the rules $\boldsymbol{d}_k^*$, $k = 1, 2, 3$, in Eq. (24) respectively have the approximate measures and the exact expectations as summarized in Table 2. Then, these values can be used to evaluate $D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}})$, which is also summarized in the table. The result suggests the rule $\boldsymbol{d}_3^*$ as the next basis rule. The iterations with different basis rules are terminated when the convergence can be concluded, i.e. further iterations do not generate any new solutions. Despite the subjectivity, such termination strategy is found to be valid from numerical experiments.

### 5.2. Example reliability block diagram

Consider the reliability block diagram in Fig. 8 adopted from [2] with the ID in Fig. 2 where $N = 8$. The system event represented by $X_9$, is defined as the connectivity between $s$ and $t$ nodes. The definitions of variables $X_n$, $D_n$, $V_n$, $n = 1, \ldots, 8$, and $V_9$ are defined as in Section 2.2. Three design alternatives are considered for each component, i.e. $Val(D_n) = \{1, 2, 3\}$, $n = 1, \ldots, 8$, and the CPMs $\mathcal{M}_{X_n} = <\mathbf{C}_{X_n}; \mathbf{p}_{X_n}>$ are constructed as

**Table 2**
$\tilde{E}[V_{N+1}|\boldsymbol{d}_k^*; \bar{\boldsymbol{d}}]$, $E[V_{N+1}|\boldsymbol{d}_k^*]$, and $D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}})$ for optimal solutions $\boldsymbol{d}_k^*$, $k = 1, \cdots, 4$.

| $k$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $\tilde{E}[V_{N+1}\|\boldsymbol{d}_k^*; \bar{\boldsymbol{d}}]$ ($\times 10^{-1}$) | 2.50 | 2.67 | 2.85 | 3.26 |
| $E[V_{N+1}\|\boldsymbol{d}_k^*]$ ($\times 10^{-2}$) | 4.58 | 5.72 | 6.72 | 10.9 |
| $D(\boldsymbol{d}_k^*; \bar{\boldsymbol{d}})$ | −1.54 | −1.76 | −1.00 | – |

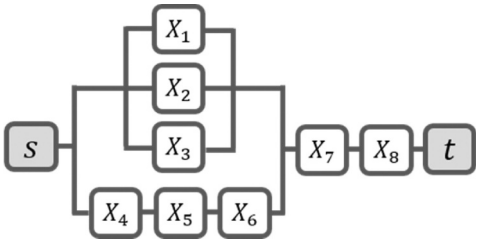


**Fig. 8.** Example reliability block diagram.

$$\mathbf{C}_{X_n} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 2 \\ 1 & 2 \\ 0 & 3 \\ 1 & 3 \end{bmatrix} \text{ for } n = 1, \cdots, 8; \quad \mathbf{p}_{X_n} = \begin{bmatrix} 0.01 \\ 0.99 \\ 0.003 \\ 0.997 \\ 0.001 \\ 0.999 \end{bmatrix} \text{ for } n = 1, 2, 4, 7;$$
$$\text{and } \mathbf{p}_{X_n} = \begin{bmatrix} 0.01 \\ 0.99 \\ 0.005 \\ 0.995 \\ 0.001 \\ 0.999 \end{bmatrix} \text{ for } n = 3, 5, 6, 8 \tag{25}$$

where the first and second columns of $\mathbf{C}_{X_n}$ respectively represent the states of $X_n$ and $D_n$. On the other hand, the CPMs for costs, $\mathcal{M}_{V_n} = <\mathbf{C}_{V_n}; \mathbf{p}_{V_n}>$ are set as

$$\mathbf{C}_{V_n} = \begin{bmatrix} 100 & 1 \\ 150 & 2 \\ 250 & 3 \end{bmatrix} \text{ for } n = 1, 2, 4, 7; \quad \mathbf{C}_{V_n} = \begin{bmatrix} 80 & 1 \\ 140 & 2 \\ 200 & 3 \end{bmatrix} \text{ for } n = 3, 5, 6, 8;$$
$$\text{and } \mathbf{p}_{X_n} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \text{ for } n = 1, \ldots, 8 \tag{26}$$

where the first and second columns of $\mathbf{C}_{V_n}$ respectively denote the assignments over $V_n$ and $D_n$.

Fig. 9 illustrates the 24 non-dominated solutions (red squares) among the total of $|Val(\boldsymbol{D})| = 3^8$ rules (blue dots). The proposed method can identify all optimal solutions (yellow asterisks) but two solutions by ten iterations, which implies that the proposed method can obtain optimization solutions only with the computational cost that would be needed to evaluate $E[V_9|\boldsymbol{d}]$ for ten $\boldsymbol{d}$'s. In Fig. 9, it is noted that the missing two solutions form eminently concave shapes with adjacent solutions, which implies that they are trivial solutions in terms of practical use. Still, the proposed method identifies the majority of the

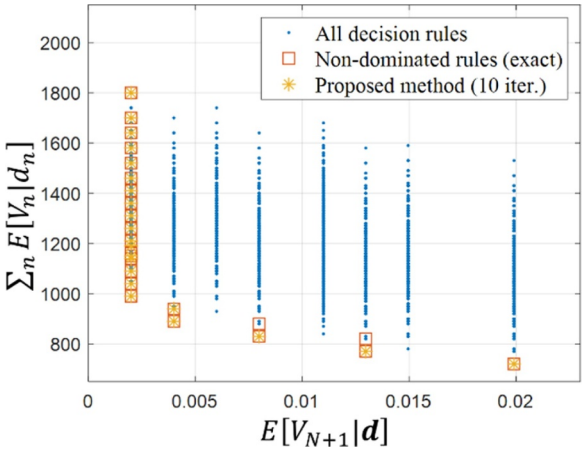


**Fig. 9.** All and optimal solutions obtained for example reliability block diagram.

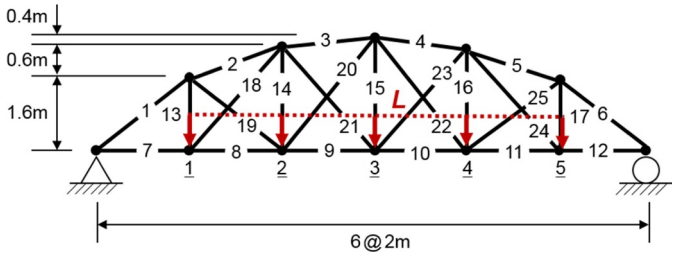


**Fig. 10.** Example truss bridge structure.

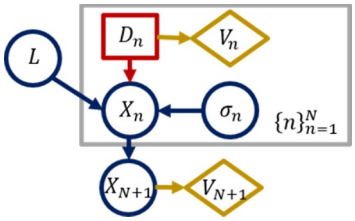


**Fig. 11.** ID for example truss bridge structure.

solutions inside the convex surface enveloping the non-dominated solutions, i.e. the solutions that cannot be identified by the exact objective function of weighted sum, which demonstrates the discussion in Section 4.3.

### 5.3. Truss bridge structure

Next, consider the truss bridge structure consisting of 25 members in Fig. 10, which is a modified version of the example in [15]. The corresponding ID is illustrated in Fig. 11. The r.v.'s $X_n$, $n = 1, \ldots, N$, and $X_{N+1}$ respectively represent the states of the $n$th member and the bridge system (0 for failure; 1 for survival) where $N = 25$. Seven cross section area values are considered as decision alternatives for each member, each of which is represented by $D_n$, $n = 1, \ldots, N$, while $V_n$ quantifies the cost for $D_n$. The costs for the areas per unit length are summarized in Table 3, based on which the CPMs for $V_n$ can be constructed as Eqs. (22) and (26), and $V_{N+1}$ is defined by Eq. (6). The r.v.'s $L$ and $\sigma_n$, $n = 1, \ldots, N$, respectively stand for the magnitude of loads applied on joints 1 through 5, and the yield stress of $n$th member.

The system failure is defined as "structural instability occurs or more than two members fail together." As a result, the CPM $\mathcal{M}_{X_{N+1}}$ includes 4779 rules each of which corresponds to a failure mode. It is noted that each failure mode corresponds to a sequential failure of members that leads to system failure, and thus, they are disjoint, e.g. the failure sequence of members {1}, {2→1}, {2→4→6}, ···. Therefore, $\mathcal{M}_{X_{N+1}}$ satisfies the condition that the rules in a CPM must be mutually exclusive.

On the other hand, it is assumed that $L$ and $\sigma_n$, $n = 1, \ldots, N$, are all statistically independent of each other and follow normal distributions. The mean and coefficient of variation of $L$ are 70 kN and 0.1 while those for $\sigma_n$ are 276 MPa and 0.05. The distribution of $L$ is discretized into 9 intervals with the width of 5 kN. The midpoints of the intervals are regarded as their representative values. In the case of $\sigma_n$, $n = 1, \ldots, N$, they are not discretized but marginalized before optimization.

**Table 3**
Cost per unit length for each cross section area option.

| Area ( $\times$ $10^{-4}$ m$^2$) | Cost/length (m) |
|---|---|
| 12.0 | 10.0 |
| 12.5 | 30.0 |
| 13.0 | 45.0 |
| 13.5 | 55.0 |
| 14.0 | 60.0 |
| 14.5 | 62.5 |
| 15.0 | 65.0 |

The probabilities of component events constituting each failure mode can be computed by performing structural analysis for each value of $l \in Val(L)$ and $d_n \in Val(D_n)$. For example, consider the failure mode {2→1}. The structural analysis is first performed to evaluate the demands $R_2$ and $R_1$ on members 2 and 1. The analysis is performed again after removing member 2 to obtain the demands after load re-distribution, i.e. $R_1'$ and $R_n'$ for members 1 and $n = 3, \ldots, 25$. The probability of the failure mode is then evaluated as

$$P\left(\frac{R_1}{A(d_1)} \le \sigma_1 \le \frac{R_1'}{A(d_1)}\right) \cdot P\left(\sigma_2 \le \frac{R_2}{A(d_2)}\right) \cdot \prod_{n=3}^{N} P\left(\sigma_n \ge \frac{R_n'}{A(d_n)}\right) \tag{27}$$

where $A(d_n)$ is the cross section area when $d_n$ is chosen. The marginalization of $\sigma_n$ is equivalent to calculating the probability values in Eq. (27).

Fig. 12 compares the optimization results by the proposed method and GA. As illustrated in Fig. 12(a), the proposed method can provide the overall shape of Pareto surface with only two iterations while the number of the obtained solutions is not large. On the other hand, GA can provide the contour only after the solutions finally reach convergence, which is also not straightforward to conclude. Moreover, with only 15 iterations, the proposed method can provide a result which is comparable to that obtained by GA after 500 generations. The solutions obtained at each iteration, can also be used to numerically check the condition of Eq. (9) as discussed in Section 4. $E[V_{N+1}|\boldsymbol{d}^*]$ and $\tilde{E}[V_{N+1}|\boldsymbol{d}^*;\tilde{\boldsymbol{d}}]$ for the optimal solutions $\boldsymbol{d}^* \in \boldsymbol{D}^*$ obtained at the second iteration are plotted in Fig. 13 where the positive correlation between the two quantities are observed, i.e. Eq. (9) holds and the proposed method is valid for this problem.

### 5.4. Sioux falls benchmark network

In this example, the seismic capacities of a transportation network are optimized under a hypothetical earthquake hazards. The topology of the hypothetical network is created based on Sioux Falls benchmark example [5,16]. as illustrated in Fig. 14. The ID of the optimization problem is constructed as shown in Fig. 15. The state of $n$th link (0 for failure, and 1 for survival) is represented by r.v. $X_n$, $n = 1, \ldots, N$ with $N = 76$ while $X_{N+1}$ denotes the system state. Due to the large number of links, it is computationally intractable to enumerate all link-sets and cut-sets. Thus, 1038 cut-sets among the total of 5000 link- and cut-sets identified by the recursive decomposition algorithm [17], are used in this example to compute the lower bound of the system failure probability. The lower bound on system failure probability is optimized from the subset of events with the CPM for $V_{N+1}$ in Eq. (6), which is one of the advantages of the MBN that allows consistent operations for exact and approximate inferences [5].

The earthquake hazards are represented by two r.v.'s $M$ and $L$ which respectively stand for the magnitude and the location of the fault rupture of an earthquake event. The magnitude $M$ is assumed to follow the truncated exponential distribution [18]. i.e.

$$f_M(m) = \begin{cases} \frac{\beta \exp[-\beta(m-m_0)]}{1-\exp[-\beta(m-m_0)]}, & \text{for } m_0 \le m \le m_p \\ 0, & \text{otherwise} \end{cases} \tag{28}$$

where the parameters $\beta$, $m_0$, and $m_p$ are respectively set as 0.76, 6.0, and 8.5. The ten locations illustrated by red squares in Fig. 14 are represented by the r.v. $L$ with $P(l^k) = 0.1$, $k = 1, \ldots, 10$. For optimization, the distribution of $M$ is discretized into five intervals with identical widths.

Four levels of seismic capacity are considered for each link, i.e. $|Val(D_n)| = 4$, which are given in terms of peak ground acceleration (PGA) as 0.8, 0.9, 1.0, and 1.1$g$ where $g$ is the acceleration of the gravity. These decision alternatives are respectively associated with the

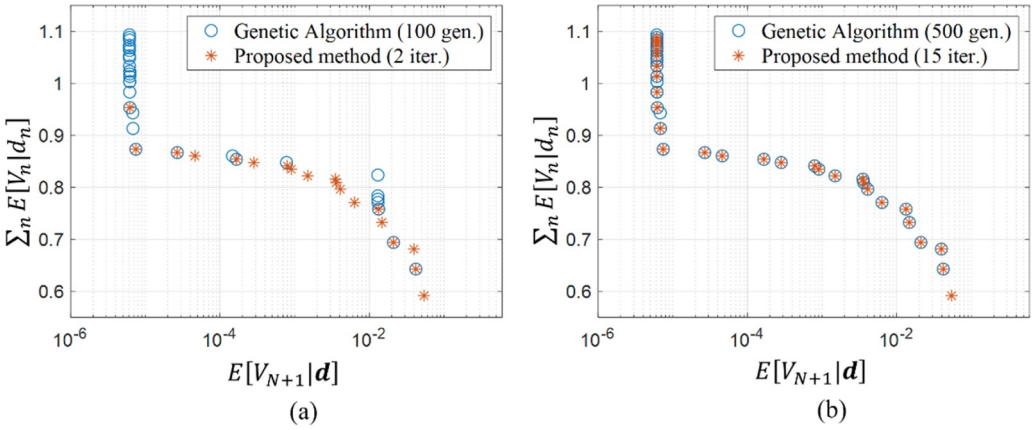


**Fig. 12.** Comparison between the results for truss bridge structure by (a) 100 generations of GA and 2 iterations of the proposed method, and (b) 500 generations of GA and 15 iterations of the proposed method.

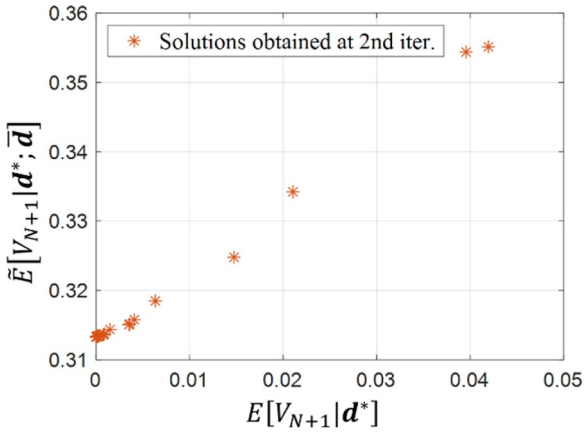


**Fig. 13.** $E[V_{N+1}|\boldsymbol{d}^*]$ and $\tilde{E}[V_{N+1}|\boldsymbol{d}^*;\bar{\boldsymbol{d}}]$ for solutions $\boldsymbol{D}^*$ obtained at the second iteration for example truss bridge.

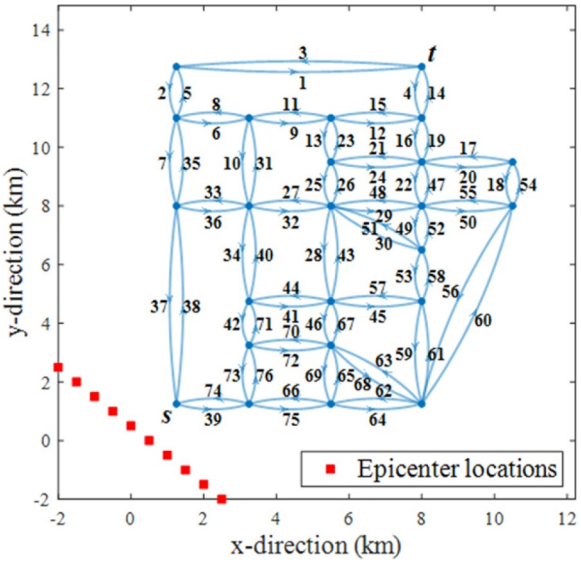


**Fig. 14.** Sioux Falls network.

cost of 50, 80, 120, and 170, from which the CPMs for $V_n$ can be constructed as illustrated in Eqs. (22) and (26). For each assignment of $m \in Val(M)$ and $l \in Val(L)$, the PGA $A_n$ that $n$th link experiences is evaluated from the attenuation law in [19]. i.e.

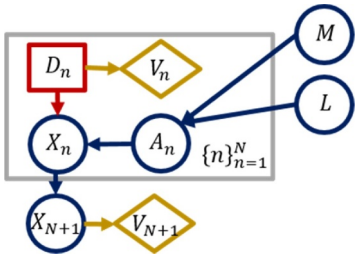


**Fig. 15.** ID for Sioux Falls network.

$$\begin{aligned}\ln(A_n) = &-3.512 + 0.904m - 1.328\ln\sqrt{[R_n(l)]^2 + [0.147\exp(0.647m)]^2} \\ &+ [1.125 - 0.112\ln(R_n(l)) - 0.0957m]F \\ &+ [0.440 - 0.171\ln(R_n(l))]S_{SR} + [0.405 - 0.222\ln(R_n(l))]S_{HR} \\ &+ \epsilon \end{aligned} \tag{29}$$

where $A_n$ has the unit of $g$ and $R_n(l)$ is the distance between $n$th link and the location of fault rupture, which is determined by $l$. The midpoint of the two end nodes is selected as the representative location of each link. The constant $F$ describes the style of faulting while $S_{SR}$ and $S_{HR}$ reflect the local site conditions. These constants are all set as zero, assuming strike-slip faulting and firm soil. The r.v. $\epsilon$ is a random error term that follows the normal distribution with zero mean and the standard deviation $\sigma_\epsilon$ which is determined as

$$\sigma_\epsilon = \begin{cases} 0.55, & \text{if } A_n < 0.068g \\ 0.173 - 0.140\ln(A_H), & \text{if } 0.068g \le A_n \le 0.21g \\ 0.39, & \text{otherwise} \end{cases} \tag{30}$$

Then, the PMFs $P(X_n|M, L, D_n)$, $n = 1, \ldots, N$, are quantified as

$$P(X_n|m, l, d_n) = \begin{cases} \Phi\left(-\frac{\ln(A_n) - \ln C(d_n)}{\sigma_\epsilon}\right), & \text{if } X_n = 0 \\ \Phi\left(\frac{\ln(A_n) - \ln C(d_n)}{\sigma_\epsilon}\right), & \text{if } X_n = 1 \end{cases} \tag{31}$$

where $A_n$ and $\sigma_\epsilon$ are determined from the assignments $m$ and $l$; $C(d_n)$ is the seismic capacity when $d_n$ is chosen; and $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. The evaluation of Eq. (31) is equivalent to marginalizing $A_n$, $n = 1, \ldots, N$.

Fig. 16(a) confirms that the proposed method can identify the contour of the Pareto surface after utilizing only two basis decision rules, while GA fails to give a comparable result even after 1000 generations. Fig. 16(b) compares the results by the proposed method with 50 iterations and GA with 10,000 generations in which GA still fails to converge. The exact and proxy quantities, i.e. lower bounds of $E[V_{N+1}|\boldsymbol{d}^*]$ and $\tilde{E}[V_{N+1}|\boldsymbol{d}^*;\bar{\boldsymbol{d}}]$, for the optimal rules $\boldsymbol{d}^*$ identified with second basis decision rule are plotted in Fig. 17, confirming that they

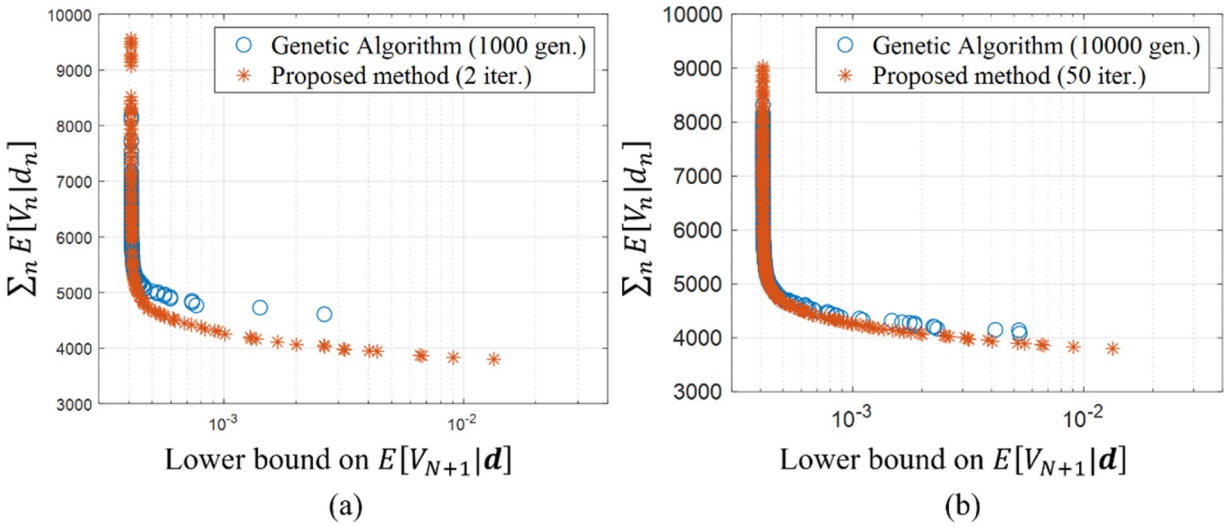


**Fig. 16.** Solutions by GA and the proposed method for Sioux Falls network with respectively (a) 1000 generations and 2 iterations, and (b) 10,000 generations and 50 iterations.

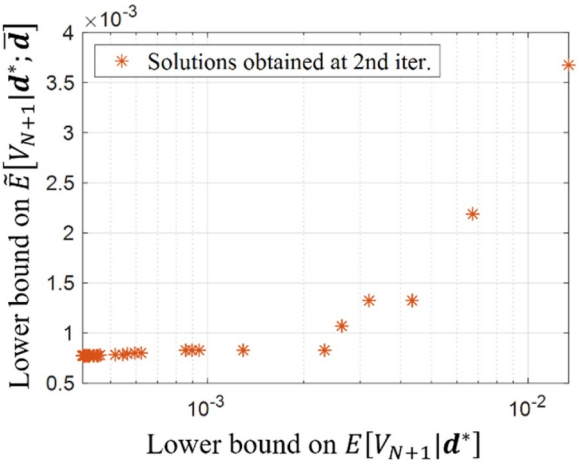


**Fig. 17.** $E[V_{N+1}|\boldsymbol{d}^*]$ and $\tilde{E}[V_{N+1}|\boldsymbol{d}^*;\bar{\boldsymbol{d}}]$ for obtained solutions $\boldsymbol{D}^*$ at second iteration for Sioux Falls network.

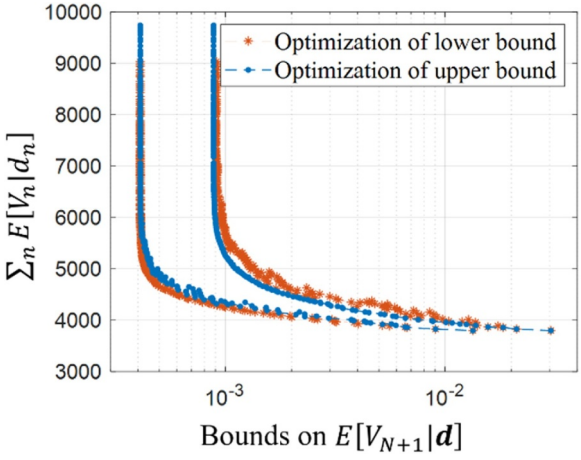


**Fig. 18.** Non-dominated solutions by optimizing lower (red squares) and upper (blue circles) bounds of system failure probability of Sioux Falls network.

generally show positive correlation and the proposed methodology is found to be valid for this problem.

The optimization of the lower bound on the system failure probability is equivalent to maximizing the upper bound of system survival probability. When a sufficient number of rules are quantified so that the width of bounds is narrow enough, the optimization results should also be similar to the solutions obtained by utilizing link-sets, i.e. minimization of the upper bound on system failure probability or equivalently, maximization of the lower bound on system survival probability. Fig. 18 compares the upper and lower bounds of $E[V_{N+1}|\boldsymbol{d}]$ that are obtained respectively by minimizing its lower bound (red asterisks) and upper bound (blue circles). The differences between the two Pareto contours are marginal, which demonstrates that the proposed scheme is applicable even when CPMs are not exhaustive.

## 6. Conclusions

In order to facilitate probabilistic multi-objective optimization of discrete Influence Diagram (ID) describing large-scale complex systems, a new optimization scheme using a proxy objective function and an alternative probabilistic modeling of complex systems was proposed. The ID representing system events is often characterized by the converging structure between the random variables (r.v.'s) standing for component events and that for system event. Such structure raises difficulties not only in quantifying probability mass functions (PMFs) in ID, but also in optimizing the sum of expectations of utility variables.

In regards to quantification, the PMF of the system event conditioned on component events should be quantified, for which the number of required parameters exponentially increases as the number of components increases. In this paper, this issue was successfully addressed by employing the matrix-based Bayesian network (MBN) and extending it for ID, which was recently proposed as an alternative data structure for the PMFs of discrete Bayesian network (BN) [5]. Using the MBN, system events can be efficiently quantified by exploiting the regularity in their definitions as demonstrated in the numerical examples. Moreover, as illustrated in Section 5.4, when a system event cannot be completely quantified due to the large number of components and its complex definition, an approximate inference can be performed by the same inference methodologies as exact inferences in which the

CPMs with only a subset of events are instead utilized. This allows the discrete BN and ID to handle even larger systems.

On the other hand, regarding optimization, the converging structure makes the decision variables, which are connected to each component, become strategically relevant to each other when optimizing the expectation of utility variable that is connected to the system node to represent the system performance. In other words, the decision variables cannot be optimized separately, and thus, all possible combinations of the assignments over decision variables need to be considered whose number also exponentially increases in regards to that of components. In order to address this issue, a proxy objective function was proposed by which the optimization problem can be decomposed with respect to each decision variable. The conditions for the equivalence between the exact and proxy functions, were also derived. It was shown that the proposed method should be valid for a wide class of problems. Moreover, in order to compensate some degree of errors caused by the approximation, empirical strategies were suggested for implementation and iteration of the proposed method. A way to numerically check the condition for the equivalence between the two objective functions, i.e. the positive correlation between the two functions, was also proposed so that the validity of the proposed method can be confirmed for a given problem.

Another advantage of the proposed proxy function is that the function enables us to analytically evaluate a set of non-dominated solutions for multiple objectives by optimizing their weighted sum. Interestingly, while the objective function described as the weighted sum cannot identify all non-dominated solutions, the approximation by the proposed method allows us to identify a larger set of such solutions. Moreover, the proxy objective function also enables analytical evaluation of suboptimal solutions, i.e. those with the smallest objective values of second, third, etc., by which the loss of solutions based on the weighted sum formulation can be further compensated.

In this paper, two objectives of cost and system failure probability were considered, and the numerical examples demonstrate the accuracy and efficiency of the proposed methodology. The proposed method is able to not only identify a sufficient number of non-dominated solutions with a small number of iterations, but also provide the overall contour at the very early stage of iterations. The proposed methodology can be expanded to incorporate other definitions of system events, e.g. flow analysis of a bridge network or more complex structures such as a cable-stayed bridge, and other types of objectives, e.g. travel time in a transportation network or socioeconomic cost entailed by decision alternatives. On the other hand, the MBN quantification scheme can be further developed by introducing advanced encoding methods. This would enable us to overcome the role of "−1" state that can partition the event space only into hypercubes.

## CRediT authorship contribution statement

**Ji-Eun Byun:** Conceptualization, Methodology, Software, Validation, Writing - original draft, Visualization. **Junho Song:** Conceptualization, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition.

## Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

This research was supported by a grant (20SCIP-B146946-03) from Smart Civil Infrastructure Research Program funded by Ministry of Land, Infrastructure and Transport of Korean government. The second author is supported by the Institute of Construction and Environmental Engineering at Seoul National University.

## References

[1] Koller D, Friedman N. Probabilistic graphical models: principles and techniques. Cambridge: The MIT Press; 2009.
[2] Bensi M, Der Kiureghian A, Straub D. Efficient bayesian network modeling of systems. Reliab Eng Syst Saf 2013;112(1):200–13.
[3] Tien I, Der Kiureghian A. Algorithms for bayesian network modeling and reliability assessment of infrastructure systems. Reliab Eng Syst Saf 2016;156(1):134–47.
[4] Zhang NL, Poole D. Exploiting causal independence in bayesian network inference. J Artif Intell Res 1996;5(1):301–28.
[5] Byun J-E, Zwirglmaier K, Straub D, Song J. Matrix-based bayesian network for efficient memory storage and flexible inference. Reliab Eng Syst Saf 2019;185(1):533–45.
[6] Olmsted SM. On representing and solving decision problems. Stanford University; 1984.
[7] Shachter RD. Probabilistic inference and influence diagrams. Oper Res 1988;36(4):589–604.
[8] Jensen FV, Vomlelov M. Unconstrained influence diagrams. Proceedings of the eighteenth conference on uncertainty in artificial intelligence. 2002. p. 234–41.
[9] Bensi M, Der Kiureghian A, Straub D. Framework for post-earthquake risk assessment and decision making for infrastructure systems. ASCE-ASME J. Risk Uncertainty Eng. Syst. Part A 2014;1(1):04014003.
[10] Memarzadeh M, Pozzi M. Value of information in sequential decision making: component inspection, permanent monitoring and system-level scheduling. Reliab. Eng. Syst. Saf. 2016;154(1):137–51.
[11] Nishijima K, Maes MA, Goyet J, Faber MH. Constrained optimization of component reliabilities in complex systems. Struct. Saf. 2009;31(2):168–78.
[12] Diehl M, Haimes YY. Influence diagrams with multiple objectives and tradeoff analysis. IEEE Trans. Syst. Man Cybern. Part A 2004;34(3):293–304.
[13] Yang DY, Frangopol DM. Probabilistic optimization framework for inspection/repair planning of fatigue-critical details using dynamic bayesian networks. Comput. Struct. 2018;198(1):40–50.
[14] Darwiche A. A differential approach to inference in bayesian networks. J. ACM 2003;26(6):826–31.
[15] Kim D-S, Ok SY, Song J. System reliability analysis using dominant failure modes identified by selective searching technique. Reliab Eng Syst Saf 2013;119(1):316–31.
[16] Lee YJ, Song J, Gardoni P, Lim HW. Post-hazard flow capacity of bridge transportation network considering structural deterioration of bridges. Struct Infrastruct Eng 2011;7(7–8):509–21.
[17] Li J, He J. A recursive decomposition algorithm for network seismic reliability evaluation. Earthq Eng Struct Dyn 2002;31(8):1525–39.
[18] Cosentino P, Ficarra V, Luzio D. Truncated exponential frequency-magnitude relationship in earthquake statistics. Bull Seismol Soc Am 1977;67(6):1615–23.
[19] Campbell KW. Empirical near-source attenuation relationships for horizontal and vertical components of peak ground acceleration, peak ground velocity, and pseudo-absolute acceleration response spectra. Seismol Res Lett 1997;68(1):154–79.