

Pareto surface construction for multi-objective optimization under uncertainty

Chen Liang¹ · Sankaran Mahadevan¹

Received: 20 October 2015 / Revised: 19 July 2016 / Accepted: 22 September 2016 / Published online: 2 December 2016
© Springer-Verlag Berlin Heidelberg 2016

Abstract This paper presents a novel approach for multi-objective optimization under both aleatory and epistemic sources of uncertainty. Given paired samples of the inputs and outputs from the system analysis model, a Bayesian network (BN) is built to represent the joint probability distribution of the inputs and outputs. In each design iteration, the optimizer provides the values of the design variables to the BN, and copula-based sampling is used to rapidly generate samples of the output variables conditioned on the input values. Samples from the conditional distributions are used to evaluate the objectives and constraints, which are fed back to the optimizer for further iteration. The proposed approach is formulated in the context of reliability-based design optimization (RBDO). The joint probability of multiple objectives and constraints is included in the formulation. The Bayesian network along with conditional sampling is exploited to select training points that enable effective construction of the Pareto front. A vehicle side impact problem is employed to demonstrate the proposed methodology.

Keywords Multi-objective optimization · Uncertainty quantification · Bayesian network · Gaussian copula

Nomenclature

x design variables
 p non-design variables
 lb_x, ub_x lower and upper bounds of x
 f_i i^{th} deterministic objective

g_i i^{th} deterministic constraint
 μ_{f_i} mean of the i^{th} objective
 P_{jt} joint probability constraint
 CDF cumulative distribution function
 PDF probability density function
 DV design variable
 UV uncertain variable
 obj objective
 $constr$ constraint
 $F_{i}(x_i)$ marginal CDF of the i^{th} random variable x_i
 C copula function
 Φ^{-1} inverse CDF of a standard normal random variable
 R covariance matrix of Gaussian copula
 I identity matrix
 L likelihood function
 L_{ab} Abdomen load
 Rib_u Upper rib deflection
 Rib_m Middle rib deflection
 Rib_l Lower rib deflection
 VC_u Upper viscous criteria
 VC_m Middle viscous criteria
 VC_l Lower viscous criteria
 F_{pb} Pubic force
 Con_i i^{th} constraint of the vehicle side impact example
 $Crit$ criterion of the i^{th} constraint

1 Introduction

In multi-objective optimization (MOO) with competing objectives, the multiple solutions are often characterized through a Pareto surface, which is a series of designs describing the tradeoff among different objectives. The decision maker will select the appropriate design alternative based on his/her preferences on the objectives (Marler and Arora 2004). Four types

✉ Sankaran Mahadevan
sankaran.mahadevan@vanderbilt.edu

¹ Department of Civil and Environmental Engineering, Vanderbilt University, Nashville, TN, USA

of approaches have been studied in the literature to construct the Pareto surface: weighted sum, goal programming, constraint-based methods, and genetic algorithm

The weighted sum approach assigns weights for each objective based on the stake-holder's preferences, and combines the multiple objectives into a single objective. The Pareto front is achieved by trying different weights for the individual objectives and performing the optimization multiple times. The goal programming approach treats each objective through an equivalent constraint (one goal), and introduces detrimental deviations for each of the goals. Then the objective is to minimize the weighted sum of the detrimental deviations. In constraint-based methods (Mavrotas 2009), one of the objective functions is selected as the only objective, and the remaining objective functions are treated as constraints. The Pareto front can be obtained by systematically varying the constraint bounds. Similarly, multiple optimizations need to be implemented. The genetic algorithm-based approach globally searches for feasible solutions, compares and ranks them based on objectives and constraints, and selects the non-dominated solutions (Deb et al. 2002). The first three approaches convert the multi-objective optimization problem into a single objective problem and solve with optimization algorithms, therefore are more efficient. Compared to the first three approaches, the genetic algorithm requires more function evaluations; however, the former three approaches are more likely to result in suboptimal solutions.

The presence of input uncertainty and model errors introduces uncertainty in the estimation of the system model outputs. As a result, optimization under uncertainty (OUU) requires an extra loop of uncertainty quantification (UQ) or reliability assessment in each optimization iteration; that is, at each design iteration, the output distributions (or probabilities of satisfying constraint thresholds) need to be evaluated given the design variable values. Such stochastic optimization formulation often suffers from intensive computational effort. For example, aero-elastic wing analysis which contains finite element and CFD analyses may take several hours to complete even one full analysis, thus making such double loop implementation unaffordable.

Therefore, surrogate modeling techniques, which replace the expensive physics code with an inexpensive model for UQ and reliability analysis, have been studied for optimization under uncertainty. However, most surrogate modeling approaches suffer from the curse of dimensionality and may be inaccurate for modeling a system with a large number of input and output variables. Furthermore, multiple outputs need to be considered in multi-objective optimization. If the surrogate models are built separately for individual outputs, the correlations between the outputs are likely to be missed. To overcome this challenge, surrogate modeling that considers output dependence has been proposed using techniques such as co-kriging (Knotters et al. 1995). However, the size of the

co-kriging covariance matrix grows rapidly as the number of outputs considered increases; thus one can incorporate dependence between only a small number of output variables at present. Therefore, it is important to develop effective OUU methods that can handle a large number of design variables and multiple objectives, while still preserving the correlations between the objectives.

Three types of uncertainty sources need to be considered in design optimization: physical variability, data uncertainty and model uncertainty. Physical variability (aleatory uncertainty) in loads, system properties, etc., is irreducible and is commonly represented through probability distributions. Data uncertainty (epistemic) may be caused by sparse and/or imprecise data, and can be reduced by collecting more information. Model uncertainty (epistemic) arises from the model used to approximate the physics, and can be attributed to three types of sources: uncertain model parameters (due to limited data), numerical errors (i.e., solution approximations due to limited computational resources), and model form error (due to the assumptions made in the model) (Rebba et al. 2006). The propagation of aleatory uncertainty is well-studied in the literature, and can be accomplished by Monte Carlo sampling or First/Second-Order Reliability Methods (FORM/SORM) (Haldar and Mahadevan 2000). Epistemic uncertainty (lack of knowledge) is an active research topic, and needs careful treatment due to its variety of sources and representation formats.

Data uncertainty due to sparse or interval data has been typically represented by p-boxes (Ferson et al. 2007), Bayesian approaches (Sankararaman 2012) including family of distributions instead of a single distribution (Zaman et al. 2011), non-parametric likelihood-based distribution (Sankararaman and Mahadevan 2011), and non-probabilistic techniques such as evidence theory (Guo and Du 2009), fuzzy sets (Du et al. 2006), imprecise probabilities (Zhang and Huang 2009) and possibility theory (Du et al. 2006). Model errors (Riley and Grandhi 2011; Mahadevan and Liang 2011; Ling et al. 2014) are represented either as random variables or random processes, sometimes as functions of the input, and combined with the model prediction to give a 'corrected' model output.

Integration of uncertainty in design optimization has been investigated in two directions: (1) reliability-based design optimization (RBDO) and (2) robust design optimization (RDO). The RBDO formulation aims at finding the optimal solution that satisfies the constraints at desired probability levels, whereas the RDO formulation aims at a design that is insensitive to uncertainties. This paper addresses multi-objective optimization in the context of RBDO; however, the proposed approach can also be extended to RDO problems. Reliability assessment, which needs to calculate the probability of the output being less (or greater) than a threshold, is often applied to individual outputs. However, given a

set of design values, due to common uncertainty sources propagating through the model, the outputs are inherently correlated with each other. To enhance the system-level reliability, the joint probability of success (or failure) in meeting the reliability constraints should be introduced in the optimization formulation. This requires the consideration of output dependencies. Inclusion of dependence between the objectives has been proposed in (Rangavajhala and Mahadevan 2011), and using the joint probability as constraint has been considered in (Zhang et al. 2002). Both studies use first-order approximation and consider the input variability as the only uncertainty source. Joint probabilities of only 2 objectives in (Rangavajhala and Mahadevan 2011) and 3 objectives in (Zhang et al. 2002) are considered. As the number of output variables increases, the accuracy of the first-order approximation gets worse, whereas the number of function evaluations increases many times more than the number of variables (Liang and Mahadevan 2013). Thus a more efficient and accurate method is essential to evaluate the joint probability for a large number of variables. Towards this end, a novel concept of surrogate modeling based on the Bayesian network and copula-based sampling is proposed in this paper.

The Bayesian network is a powerful tool to incorporate different sources of uncertainty and support several applications such as model calibration (Liang and Mahadevan 2015), model validation (Mahadevan and Rebba 2005), diagnosis and prognosis (Jiang and Mahadevan 2008), reliability assessment (Zhang and Mahadevan 2000) and uncertainty quantification (Jiang and Mahadevan 2009). In general, these applications can be classified as two types of problems: inverse problem (Bayesian inference) and forward problem (uncertainty propagation). The Bayesian network supports both types of problems. Given paired samples of the inputs and outputs from a model, the BN is built to represent the joint probability distribution of the inputs and outputs through a directed acyclic graph with marginal distributions of the individual variables and the conditional probabilities between them. The BN thus functions as a probabilistic surrogate model: given specific values of the input variables, we can obtain the probability distribution of the output variables (forward problem), or given observed or desired values of the outputs, we can obtain the posterior distributions of unknown inputs (inverse problem). Both abilities of the BN are exploited in this paper.

In each design iteration, the optimizer generates a new set of design values and sends them to the Bayesian network. Given the value of the design variables, the uncertainty in the outputs can be estimated by forward propagating the uncertain variables through the BN. The resulting output is essentially the conditional joint probability distribution of the outputs given the design variable values of the inputs. Since different uncertain variables may have different types of marginal distributions, analytical calculation of this conditional

distribution is not easy. The conditional distribution estimation is often accomplished by Markov Chain Monte Carlo (MCMC) sampling techniques such as Metropolis-Hastings sampling (Chib and Greenberg 1995), slice sampling (Neal 2003), Gibbs sampling (Casella and George 1992), etc. However, MCMC sampling is quite expensive, especially when the dimension of the joint distribution increases. Considering that the conditional joint distribution will need to be estimated many times during the optimization, using tools that are efficient in sampling is significant. To meet this need, a vine copula-based sampling technique is explored in this research.

Graphical vine models are introduced in (Cooke 1997) and (Bedford and Cooke 2001). Regular vines are graphical structures for identifying bivariate and conditional bivariate joint distributions which uniquely determined a joint distribution (Kurowicka and Cooke 2010). In other words, a multivariate joint distribution can be decomposed as a series of bivariate and conditional bivariate distributions, from which samples of the variables can be easily generated by assuming a copula function. A copula is a function that relates the joint CDF of multiple variables to their marginal CDFs (Nelsen 1999). Copulas have been used in reliability analysis and RBDO for correlated (Noh et al. 2008) and non-Gaussian (Choi et al. 2010) input random variables.

Generating samples from a Bayesian network using a vine copula-based approach is quite general, but it can be computationally expensive for high-dimensional problems. However, if a joint Gaussian copula is assumed, then the conditional joint updating of the BN can be accomplished analytically (Kurowicka and Cooke 2010) which is very fast.

In optimization under uncertainty, given the values of the design inputs, the copula can be conditionally sampled (Cooke and Kurowicka 2007) to estimate the conditional joint distribution of all the output variables. The concepts of Bayesian network and copula-based sampling are combined for MOO in this research, and referred to as BNC-MOO. The proposed approach is used with an optimization algorithm to accomplish design under uncertainty. Since the proposed approach is efficient in sampling, genetic algorithms can be afforded. A Non-dominated Sorting Genetic Algorithm-II (NSGA-II) (Deb et al. 2002) that specifically solves multi-objective optimization is applied for identifying the Pareto front in this research.

Since the Bayesian network is used as a surrogate model in this research, its predictive capability largely relies on the selection of useful and informative training points. Selection of training points for enhancing the performance of surrogate models in optimization,

referred to as Efficient Global Optimization (EGO), has been studied using Gaussian process surrogate models (Jones et al. 1998), where an expected improvement function is built to select the location at which new training points should be added. Previous research has only focused on the improvement of a single function. However, this is not sufficient when multiple objectives that share the same inputs need to be improved simultaneously. This is because different training points need to be added to improve different objectives. If the EGO-based approach is used, then co-Kriging will have to be adopted to properly account for dependence among the objectives; this is computationally burdensome in the presence of multiple objectives. In this paper, a novel optimal training point selection technique is proposed based on the inverse propagation capability of the Bayesian network. A sample-based ‘sculpting’ technique (Cooke et al. 2015) is exploited to selectively choose the input samples that correspond to multiple outputs in the desired region *simultaneously*. This strategy is found to be effective and efficient in constructing the Pareto surface of solutions.

The contributions of this paper are as follows:

- (1) A new concept of probabilistic surrogate modeling technique based on the Bayesian network is adopted in order to consider large numbers of input variables, and to preserve the dependence among the objectives and among the constraints.
- (2) The BNC approach is developed for multi-objective optimization under uncertainty in the context of an RBDO formulation.
- (3) A novel training point selection approach is proposed using sample-based conditioning of the BN, in order to efficiently construct the Pareto surface.

The rest of the paper is organized as follows. Section 2 briefly formulates single-objective and multi-objective optimization problems under uncertainty, in the context of RBDO. Section 3 develops two innovations: (1) the use of the BNC approach for efficient uncertainty quantification and design optimization, and (2) the use of sample-based conditioning to improve the Pareto surface. The first innovation exploits forward propagation through BN, whereas the second innovation exploits inverse propagation through the BN. Section 4 addresses data uncertainty and explains the construction of empirical distributions from sparse and interval data, to be used in the Bayesian network and optimization. An automotive side impact problem is used in Section 5 for numerical illustration of the proposed MOUU methodology. Section 6 provides concluding remarks.

2 Multi-objective optimization under uncertainty

2.1 Single objective optimization

A typical deterministic design optimization formulation can be given as follows.

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}, \mathbf{p}) & (1) \\ \text{s.t.} & & \\ & g_i(\mathbf{x}, \mathbf{p}) \leq 0, \quad i = \{1, \dots, n_g\} \\ & lb_{x_k} \leq x_k \leq ub_{x_k}, \quad k = \{1, \dots, n_x\} \end{aligned}$$

where f is the performance function or objective to be minimized; \mathbf{x} is the vector of design variables; \mathbf{p} is the vector of non-design variables (i.e., not controlled by the designer); n_g and n_x are the number of constraints and design variables, respectively; lb_{x_k} and ub_{x_k} are the lower and upper bounds of x_k . When the uncertainties in variables \mathbf{x} and \mathbf{p} are of the aleatory type, the formulation and solution approaches are rather well-established; a survey is provided in (Valdebenito and Schuëller 2010). An RBDO formulation of the above problem combining uncertainty can be given as

$$\begin{aligned} & \min_{\mu_{\mathbf{x}}, \mathbf{d}} \mu_f(\mathbf{X}, \mathbf{d}, \mathbf{P}, \mathbf{p}_d) & (2) \\ \text{s.t.} & & \\ & \text{Prob}(g_i(\mathbf{X}, \mathbf{d}, \mathbf{P}, \mathbf{p}_d) \leq 0) \geq p_i^t, \quad i = \{1, \dots, n_g\} \\ & \text{Prob}(\mathbf{X} \geq lb_{\mathbf{X}}) \geq p_{lb}^t \\ & \text{Prob}(\mathbf{X} \leq ub_{\mathbf{X}}) \geq p_{ub}^t \\ & lb_{\mathbf{d}} \leq \mathbf{d} \leq ub_{\mathbf{d}} \end{aligned}$$

where \mathbf{X} is the vector of random design variables with bounds $lb_{\mathbf{X}}$ and $ub_{\mathbf{X}}$, respectively; μ_f and $\mu_{\mathbf{X}}$ are the mean of f and \mathbf{X} , respectively; \mathbf{d} is the vector of deterministic design variables with bounds $lb_{\mathbf{d}}$ and $ub_{\mathbf{d}}$; \mathbf{P} is the vector of random non-design variables; \mathbf{p}_d is the vector of deterministic parameters. The upper case notations represent stochastic quantities, whereas the lower case notations denote deterministic quantities. p_i^t is the target reliability required for the i^{th} constraint; p_{lb}^t and p_{ub}^t are the target reliabilities for the design variable bounds. An alternate formulation for the inequality/bound constraints using the concept of feasibility robustness involves narrowing the constraint boundaries by a multiple of their respective standard deviations (Liu et al. 2011).

Evaluation of the objective and constraints in (2) can be done through Monte Carlo sampling, but it is computationally intensive to implement when the original physics codes are time-consuming. Efficient reliability approaches such as FORM and SORM can significantly reduce the effort, yet still need dozens of function evaluations for a given design value. The total number of function evaluations will accumulate as the number of design variables and iterations increases. Therefore, surrogate modeling techniques have been studied to replace the original system model with computationally

inexpensive models for the objective and constraints estimation.

Based on the above single objective optimization formulations, the multi-objective problems are formulated in Section 2.2.

2.2 Multi-objective optimization (MOO)

A generic formulation of deterministic multi-objective optimization with n_{obj} objectives may be written as:

$$\begin{aligned}
 & \min_x \left\{ f_1(\mathbf{x}, \mathbf{p}), \dots, f_{n_{obj}}(\mathbf{x}, \mathbf{p}) \right\} \\
 s.t. \quad & g_i(\mathbf{x}, \mathbf{p}) \leq 0, \quad i = 1 \dots n_{con} \\
 & \mathbf{l} \mathbf{b}_x \leq \mathbf{x} \leq \mathbf{u} \mathbf{b}_x
 \end{aligned} \tag{3}$$

where \mathbf{x} and \mathbf{p} are the design and non-design variables, and g_i ($i = 1 \dots n_{con}$) are the deterministic constraints. $\mathbf{l}b_x$ and $\mathbf{u}b_x$ represent the upper and lower bounds for the design variables \mathbf{x} . The Pareto front for such a problem indicates the tradeoff between the function values of objectives $f_i, i = 1 \dots n_{obj}$.

In the presence of input variability, data uncertainty, and model uncertainty, the outputs become random variables, resulting in stochastic objectives and constraints. (Variances of the objectives can also be included for minimization). In this scenario, the mean values $\mu_{f_1}, \dots, \mu_{f_{n_{obj}}}$ are often considered as the objectives. In the context of RBDO, deterministic constraints $g_1, \dots, g_{n_{con}}$ are rewritten with desired probabilities of satisfaction. Therefore, MOO with probabilistic constraints can be formulated as:

$$\begin{aligned}
 & \min_{\mu_X, \mathbf{d}} \begin{cases} \mu_{f_1}(\mathbf{X}, \mathbf{P}, \mathbf{d}, \mathbf{p}_d) \\ \vdots \\ \mu_{f_{n_{obj}}}(\mathbf{X}, \mathbf{P}, \mathbf{d}, \mathbf{p}_d) \end{cases} \\
 s.t. \quad & Prob(g_i(\mathbf{X}, \mathbf{P}, \mathbf{d}, \mathbf{p}_d) \leq g_i^t) \geq P_{target}, \quad i = 1 \dots n_{con} \\
 & \mathbf{l}b_X \leq \mu_X \leq \mathbf{u} \mathbf{b}_X \\
 & \mathbf{l} \mathbf{b}_d \leq \mathbf{d} \leq \mathbf{u} \mathbf{b}_d
 \end{aligned} \tag{4}$$

The joint probability of satisfying all the constraints may also be added as a constraint as:

$$Prob\left(\bigcap_{i=1}^{n_{con}} (g_i(\mathbf{X}, \mathbf{P}, \mathbf{d}, \mathbf{p}_d) \leq g_i^t)\right) \geq P_{jt} \tag{5}$$

Estimating the joint probability of multiple events using first-order approximation at was proposed in (Hohenbichler and Rackwitz 1983) and improved in (Gollwitzer and Rackwitz 1988; Smith and Mahadevan 2005). The first-order approximation was used in (Rangavajhala and Mahadevan 2011) to estimate the joint probability distribution of the objectives in MOO. The first-order approximation could become inadequate in the presence of nonlinear objectives and constraints, and when the number of objectives and

constraints is large. On the other hand, Monte Carlo sampling can be accurate, but very expensive. Therefore, a surrogate modeling strategy combining Bayesian network and copula sampling is developed in the next section, to achieve both the desired accuracy and efficiency.

3 Proposed methodology

3.1 Bayesian network

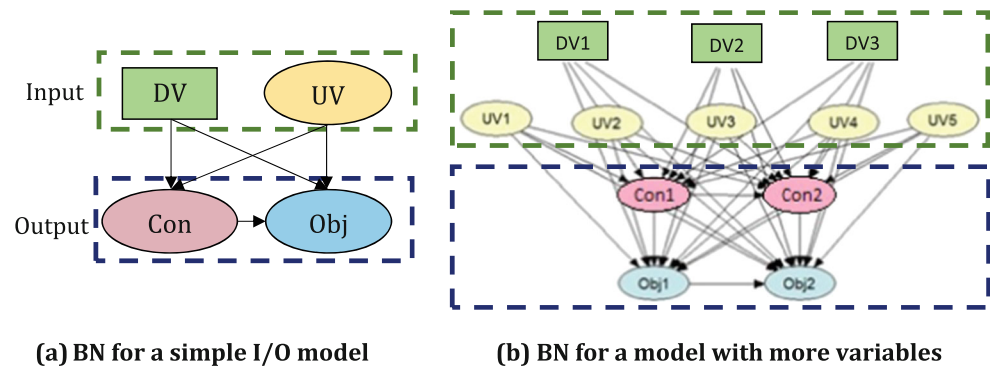
A Bayesian network is a directed acyclic graph that represents a multivariate joint probability distribution of random variables using their marginal distributions (nodes) and the conditional probabilities (arrows). The Bayesian network is capable of incorporating heterogeneous marginal distributions (e.g., continuous, discrete, binomial and empirical), and is also able to include functional relationships between the nodes if available.

For the purpose of illustration, the proposed use of the Bayesian network is shown for simple optimization problems in Fig. 1. Consider a model with two input variables and two output variables. In the context of optimization under uncertainty, one input variable is treated as design variable (*DV*), and the other is regarded as uncertain variable (*UV*). It is worth noting that, *UV* may either be a random non-design variable per se; or it may represent the variability associated with the design variable, introduced in realizing the values of the design variables in the actual system (due to manufacturing factors, for example). The 2 output variables, in the optimization setting, are used as objective variable (*Obj*) and constraint variable (*Con*). The relations among the input and output variables in such scenario are characterized by a BN as shown in Fig. 1a.

Figure 1b shows a more complicated scenario with 3 design variables (*DV1*, *DV2*, and *DV3*), which are shown in the top row of the BN. Each design variable is associated with variability (represented by *UV1*, *UV2* and *UV3*) shown in the bottom row. The variability variables have zero mean. The model input (e.g., $X_i, i = 1, 2, 3$) thus consists of two components: design variable value, and uncertain variable value (i.e., $X1 = DV1 + UV1$). The second row also includes two additional uncertain input variables *UV4* and *UV5* that are random non-design variables. The outputs are shown in the 3rd (*Con1* and *Con2*) and 4th rows (*Obj1* and *Obj2*).

The Bayesian network can be used for problems in 2 directions: (1) parameter updating (inverse) and (2) uncertainty propagation (forward), both of which need the estimation of a conditional joint distribution. As mentioned earlier, uncertainty analysis in optimization is a forward problem. In the proposed method, the Bayesian network is used as a probabilistic surrogate model that connects the input and output variables through a joint distribution. (Commonly used surrogate

Fig. 1 Bayesian network representation of optimization under uncertainty



(a) BN for a simple I/O model

(b) BN for a model with more variables

models seek to predict output *values* for given input values, whereas the BN surrogate model can be used to provide the *probability distribution* of the output for given input values).

For the purpose of illustration, consider the BN in Fig. 1b, which represents the joint distribution of the design variables and output variables, and can be written as $f(Obj1, Obj2, Constr1, Constr2, DV_i), i = 1, 2, 3$. In each optimization iteration, the design variables can be conditioned at the design values (ϕ), and the conditional joint distribution of the output variables $f(Obj1, Obj2, Constr1, Constr2 | DV_i = \phi_i), i = 1, 2, 3$ is estimated. Note that in the BN, the nodes corresponding to stochastic variables are represented by ovals. In the context of design optimization, the design variable nodes (DVs) are conditioned at specific design values in each iteration, and are therefore deterministic quantities.

Once the BN is constructed, it can be used for generating conditional samples for both the forward and inverse problems. A vine copula-based strategy is proposed, as explained next.

3.2 Copula

A copula is simply a joint CDF of marginal CDFs (Kurowicka and Roger 2006). Consider n random variables $X_1 \dots X_n$, with continuous CDFs $F_1(x_1) \dots F_n(x_n)$. The copula function related to $X_1 \dots X_n$ is defined as (6).

$$C(u_1 \dots u_n) = P[F_1(X_1) \leq u_1, \dots, F_n(X_n) \leq u_n] \quad (6)$$

The copula C contains the dependence information between the variables, and can be used to model the joint distribution of random variables. Multiple choices for copula type are available, such as Gaussian, Clayton, Frank and Gumbel copula (Nelsen 1999). In practice, Gaussian copula is widely used because of the availability of analytical solutions, which makes it much easier to model high-dimensional dependence.

Note that the Gaussian copula assumption is only for connecting the CDFs of the random variables, which is less restrictive than assuming a joint Gaussian distribution for the variables themselves. The individual variables within a Gaussian copula can have any distribution. In constructing a general BN, (1) connections between nodes (i.e. network topology) could be based on the analyst's underlying knowledge of the dependence relations within the model, whereas (2) marginal and conditional probability distributions (and other unknown connections) could be learned from data (samples). Therefore, the number of samples needs to be sufficient for an accurate BN model. Note that in the context of the optimization problems in this paper, the inputs and outputs are already defined, thus the first step is greatly simplified.

A verification approach proposed in (Hanea 2010) is adopted here to test the Gaussian copula assumption. Samples from the Gaussian copula are generated 100 times. The determinant of the normal rank correlation coefficient matrix is calculated for each set of samples, and are denoted as $detmat_i, i = 1 \dots 100$. If the determinant based on the original training samples $detmat_{ori}$ lands within 90% of the probability bounds of $detmat_i, i = 1 \dots 100$, the copula is assumed valid. Otherwise, the Gaussian copula does not hold, and other copulas need to be investigated. However, if other copulas are chosen, the sampling efficiency of the stochastic analysis will be downgraded due to the need for multiple inverse copula evaluations as explained in the Appendix, and thus the condition of BN and the sampling becomes more time-consuming. However, the computational effort in BN construction and sampling is very small in comparison to the system analysis (Liang and Mahadevan 2013), thus the BNC approach is advantageous even with non-Gaussian copula.

When the Gaussian copula assumption is not valid, other types of copula need to be pursued. However, most other copulas only have analytical forms for bi-variate dependence. Therefore, a vine-based strategy has been introduced to decompose high-dimensional dependences into bi-variate dependences, which could be modeled by any types of copula (Cooke 1997; Bedford and Cooke 2001; Kurowicka and

Cooke 2010). Details of vine-based copula are provided in Appendix.

The accuracy of the BN is also affected by the number of training samples. When the number of training samples is too small, the marginal distribution of the variables and the correlations among them may not be accurately estimated. The sufficient number of training samples can be determined by gradually adding samples and observing the change in correlation coefficients. When the change is negligible, then the number of training samples is regarded as sufficient. The number of samples chosen also depends on the computation time of the original model, and the computational resource allowed.

In summary, the three essential elements for the proposed BNC method are (1) Bayesian network, which establishes the conditional relationship among variables; (2) vine, which identifies the bivariate and conditional bivariate correlations (through) within the Bayesian network; and (3) vine copula-based conditional sampling.

3.3 Uncertainty propagation using vine copula-based sampling

Forward conditioning will be implemented in each iteration of the optimization, as illustrated in the parallel plot in Fig. 2. During each optimization iteration, the optimizer will generate values for the design variables DVs (circles in both figures). The network is updated conditioned on these values using Gaussian copula-based sampling. Samples from the conditional distribution (dashed square on the right figure) are used to estimate the objectives (mean values) and the reliability constraints (probability values).

3.4 Training point selection for Pareto surface construction

Based on the performance of the surrogate model, training points can be selectively added to improve the Pareto surface. In MOUU problems, the mean values of several objectives

need to be optimized simultaneously. As mentioned in Section 1, training point selection methods in surrogate-based optimization (e.g., EGO) mostly focus on considering a single objective, and little work has been done on multiple objectives simultaneously. In the current section, the Bayesian network is exploited through a sample-based conditioning strategy for the purpose of improving the Pareto surface. The appendix discusses the forward uncertainty propagation using the Bayesian network.

The proposed strategy for Pareto surface improvement using the Bayesian network, illustrated in Fig. 3 is as follows.

- (1) A Bayesian network is first constructed using an initial set of training points (i.e., values of input variables – both design variables and uncertain variables).
- (2) Using samples from this BN, the desired ranges of values of the obj_1 and obj_2 (as shown in Fig. 3a) are identified, and the corresponding input variable samples are identified as shown in Fig. 3b.
- (3) Then, a few of the input samples identified from Fig. 3b are selected as the new training points and the corresponding values of obj_1 , obj_2 , con_1 and con_2 are calculated with the original model.
- (4) Next, a new Bayesian network is built with both the original and additional samples, and the improved Pareto surface is constructed using the aforementioned NSGA-II algorithm.

This ‘sculpting’ strategy is both efficient and preserves the dependence among different objectives, since the new training points correspond to output samples in the desired region for all objectives simultaneously. A BN-based sculpting strategy was proposed in (Cooke et al. 2015). However, in (Cooke et al. 2015), sculpting was used to identify regions of optimum solution among the available samples generated from the BN, based on forward propagation. Whereas in this paper, sculpting is used to identify additional training samples to improve the BN, so that new regions of optimum solutions may be identified, this uses inverse propagation.

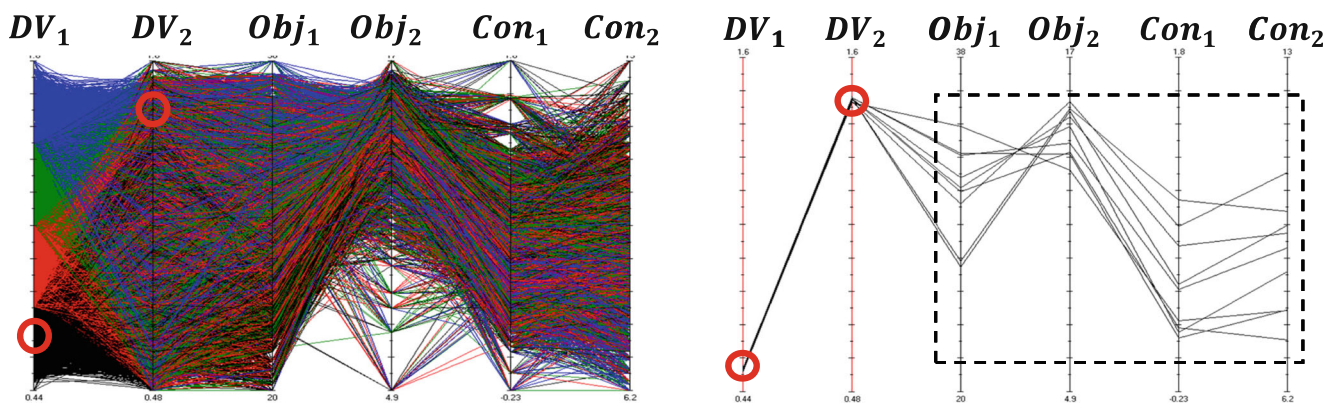


Fig. 2 Parallel coordinate representation of the model dependencies

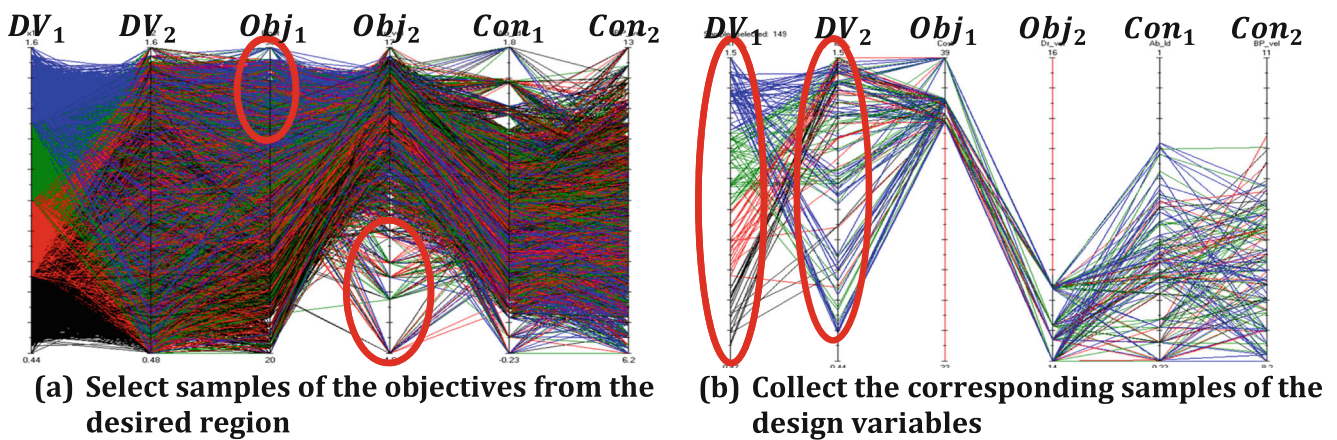


Fig. 3 Sculpting for training points selection

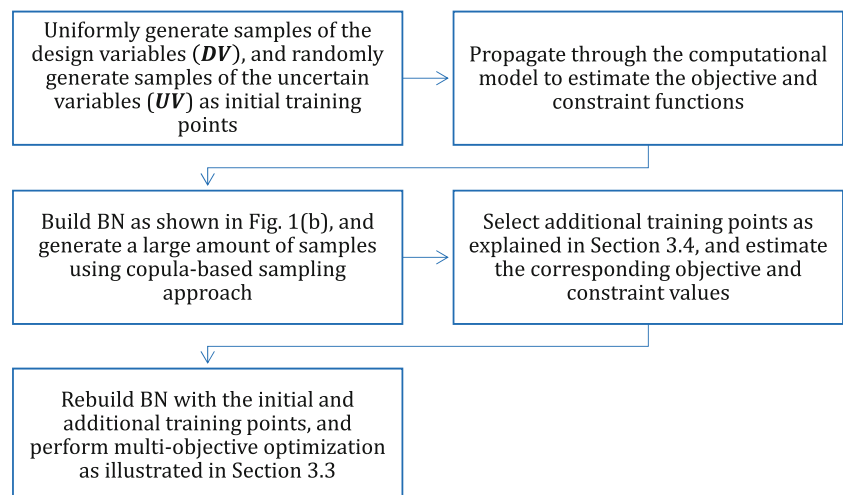
In addition, the choice of the “desired range” referred in Step (2) relies on the density of the samples of the objectives near the optimal. Since the objectives of optimization problems are usually maximization or minimization, the ranges should be set close to the poles of the objective axes. When the number of samples around the poles is high, the ranges should be set smaller (such as Obj_1); yet if the number is low, then the ranges should be increased accordingly (such as Obj_2).

The choice of the range will affect the outcome. In the numerical examples in Section 5, the mean values of the samples enclosed by the ranges are used as objective; therefore, the result is more sensitive to location of the centers of the ranges, rather than to their widths. The sensitivity analysis is out of the scope of this paper, but will be investigated in future study. The training point selection and multi-objective optimization processes are summarized in the flowchart as shown in Fig. 4.

4 Constructing empirical distributions from sparse and interval data

In the implementation of the aforementioned strategies for MOUU and Pareto surface improvement, samples of the input variables (design variables DV and uncertain variables UV in Fig. 1) need to be generated first to construct the training points for the Bayesian network. Training point values for deterministic DV can simply be generated from uniform distributions over the allowable ranges. Samples of the uncertain variables UV need to be generated from their probability distributions. When there is abundant data, the distribution types and distribution parameters of the uncertain variables may be precisely identified. However, in the presence of sparse and/or interval data, the distribution types and parameters are uncertain. Both parametric (Zaman et al. 2011), and non-parametric (Sankararaman and Mahadevan 2011) approaches have been studied to characterize the probability distributions of

Fig. 4 Flowchart for the training point selection and multi-objective optimization scheme



variables in the presence of sparse and interval data. Since the non-parametric approach is more flexible and loyal to the available data, it is used in this paper for the purpose of implementing the proposed BNC-MOO methodology. (However, parametric distributions can also be used with the proposed methodology, if those are available or preferred. The Bayesian network can accommodate both parametric and non-parametric distributions).

Suppose the available information for a random variable X is a combination of m intervals: $\{[a_1, b_1], \dots [a_m, b_m]\}$, and n point values $\{x_1, \dots x_n\}$. The domain Ω covered by the data is uniformly discretized into a finite number of points, $\theta_i, i = \{1, \dots Q\}$. Let $\mathbf{P} = \{p_1, p_2, \dots, p_Q\}$, where $p_i = f_X(x = \theta_i), i = \{1, \dots Q\}$ denotes the PDF value of X at θ_i . The PDF over the entire domain of X , i.e., $f_X(X)$, can be constructed by interpolating these PDF values. The corresponding likelihood of \mathbf{P} can be calculated as:

$$L(\mathbf{P}) \propto \left[\prod_{i=1}^n f_X(x_i | \mathbf{P}) \right] \left[\prod_{i=1}^m [F_X(b_i | \mathbf{P}) - F_X(a_i | \mathbf{P})] \right] \quad (7)$$

where m is the number of point data values, and n is the number of intervals $[a_i, b_i]$. Note that the likelihood function is only proportional to the probability of observing the given data (in the form of sparse point data and interval data). This likelihood is then maximized as in (8) shows to evaluate \mathbf{P} , i.e., the PDF values at the discretization points.

$$\begin{aligned} & \max_{\mathbf{P}} L(\mathbf{P}) \\ & \text{s.t. } p_i \geq 0 \text{ or } \forall p_i \in \mathbf{P} \\ & f_x(x) \geq 0 \text{ or } \forall x \\ & \int f_x(x) dx = 1 \end{aligned} \quad (8)$$

This approach does not assume a particular distribution type or parameters for the random variable, and represents both aleatory and epistemic uncertainty with a single nonparametric PDF. For uncertain variables with sparse and/or interval data, this strategy can be used to first construct their empirical distributions, before generating the training point samples for the Bayesian network.



Fig. 5 Vehicle side impact model (Youn et al. 2004)

5 Numerical example

A vehicle side impact model is used to demonstrate the proposed methodology (Youn et al. 2004). The model is shown in Fig. 5. A list of input and output variables is provided in Table 1. The uncertainty sources (input variability) are listed in Table 2.

The variables listed in Table 1 are used as design variables. Due to variability in the manufacturing process, variability is also assumed for each design variable as shown in Table 2. (This situation corresponds to the case shown in Fig. 1b). Note that input variables 10 and 11 were treated as design variables in (Choi et al. 2010). In this paper, they are assumed as non-design parameters, for which sparse observations (data points) plus expert-specified ranges (data intervals) are assumed to be available for the sake of illustrating the use of non-parametric empirical distributions in the proposed methodology. The likelihood-based approach is used to construct non-parametric PDFs for both variables as shown in Fig. 6.

An adequate number of training points first needs to be generated in order to construct the Bayesian network. For the side impact problem, a stepwise regression (SR) model is provided in (Youn et al. 2004), but the original data used to train the SR model are not available. Therefore, in this

Table 1 Input and output variables of the side impact model

| No. | Input | Design space | No. | Output |
|-----|---------------------------------|-----------------|-----|---------------------------------|
| 1 | B-pillar inner: x_1 | [0.5 , 1.5] | 1 | Weight |
| 2 | B-pillar reinforce: x_2 | [0.5 , 1.5] | 2 | Abdomen load: L_{ab} |
| 3 | Floor side inner: x_3 | [0.5 , 1.5] | 3 | Upper rib deflection: Rib_u |
| 4 | Cross member: x_4 | [0.5 , 1.5] | 4 | Middle rib deflection: Rib_m |
| 5 | Door beam: x_5 | [0.5 , 1.5] | 5 | Lower rib deflection: Rib_l |
| 6 | Door belt line: x_6 | [0.5 , 1.5] | 6 | Upper viscous criteria: VC_u |
| 7 | Roof rail: x_7 | [0.5 , 1.5] | 7 | Middle viscous criteria: VC_m |
| 8 | Mat. of B-pillar inner: x_8 | [0.192 , 0.345] | 8 | Lower viscous criteria: VC_l |
| 9 | Mat. of floor side inner: x_9 | [0.192 , 0.345] | 9 | Pubic force: F_{pb} |

Table 2 Uncertainty sources of the model

| No. | Input | Uncertainty type |
|-----|--------------------------|---|
| 1 | B-pillar inner | $N(0, 0.03)$ |
| 2 | B-pillar reinforce | $N(0, 0.03)$ |
| 3 | Floor side inner | $N(0, 0.03)$ |
| 4 | Cross member | $N(0, 0.03)$ |
| 5 | Door beam | $N(0, 0.03)$ |
| 6 | Door belt line | $N(0, 0.03)$ |
| 7 | Roof rail | $N(0, 0.03)$ |
| 8 | Mat. of B-pillar inner | $N(0, 0.03)$ |
| 9 | Mat. of floor side inner | $N(0, 0.03)$ |
| 10 | Barrier height | Data: 4, -8, 3.5, -0.7, 0.1, 12, [-25,20] [-30,22] [-15,31] [-28,28] |
| 11 | Barrier hitting | Data: 3, -2, 1, 0, -0.5, 0.3, [-4,5] [-8,10] [-10,7] [-0.1,1] |

research, the SR model is treated as the “original” model for the sake of illustration, from which input samples are generated to calculate the output. And then the input, uncertain and output variable values are used to train the BN.

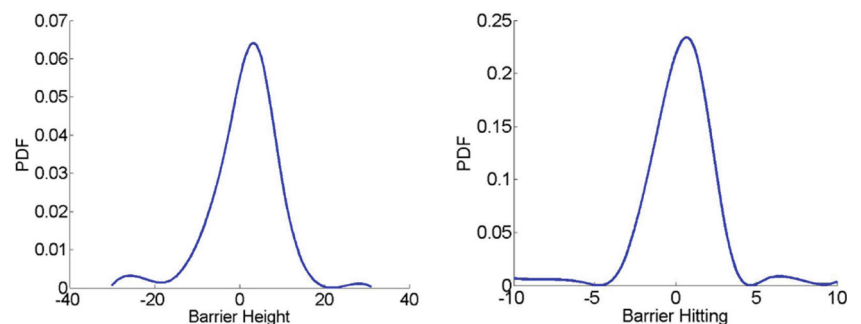
The connections (topology) between the nodes can be inferred from the SR model. At first, 100 samples are generated using Latin Hypercube sampling to train the BN. The correlation coefficient matrix is calculated based on the training samples. Then 15 more samples are added and the correlation coefficient matrix is recalculated. The differences between the corresponding entries of the two matrices are quite large. Therefore, more samples are added till the differences are acceptable. A total of 15 additional samples are generated and added to the original training samples to build a new BN. The determinant calculated from the 115 training samples is $det_{tm}^{115} = 3.14 * 10^{-15}$. Another 100 sets of samples are generated from this Gaussian copula, and the determinants of their correlation coefficient matrices are calculated. The mean and standard deviation of the determinants are $1.80 * 10^{-15}$ and $2.17 * 10^{-15}$, and det_{tm}^{115} lands in the 90% bounds. Therefore, for this example, an accurate BN can be constructed using 115 samples.

Next, the 115 samples of the design variables, combined with other uncertain variables, are propagated through the SR model. Samples of the inputs and outputs are then used to build the Bayesian network as shown in Fig. 7. This BN is a

probabilistic surrogate model of the side impact problem, i.e., for given values of some variables, it provides the joint probability distribution of the other variables.

Three optimization cases are analyzed. Case I is a single objective optimization problem. Its purpose is to verify the accuracy of the proposed BNC approach with the original SR model. The other two cases consider multi-objective optimization. Table 3 shows the correlation coefficients between the output variables. It can be seen that the correlation coefficient between the first two objectives (i.e., weight and door velocity) is -0.34 , which indicates a competing relationship between the two objectives. To simultaneously optimize these two competing quantities, a multi-objective optimization formulation from (Rangavajhala and Mahadevan 2011) is adopted, in which mean values of car weight and door velocity are minimized simultaneously. The Pareto front needs to be created to address the trade-off between the two objectives. Specifications of different cases are provided as follows:

Case I: Single objective RBDO. In this case, the mean of the car weight is used as the sole objective. The probabilities of all other 10 outputs being greater than 0.99 individually are used as reliability constraints. This is the ϵ -constraint approach. The optimization is solved with both the original SR model and the proposed BNC approach. The purpose of conducting this case

Fig. 6 Non-parametric PDFs for uncertain parameters

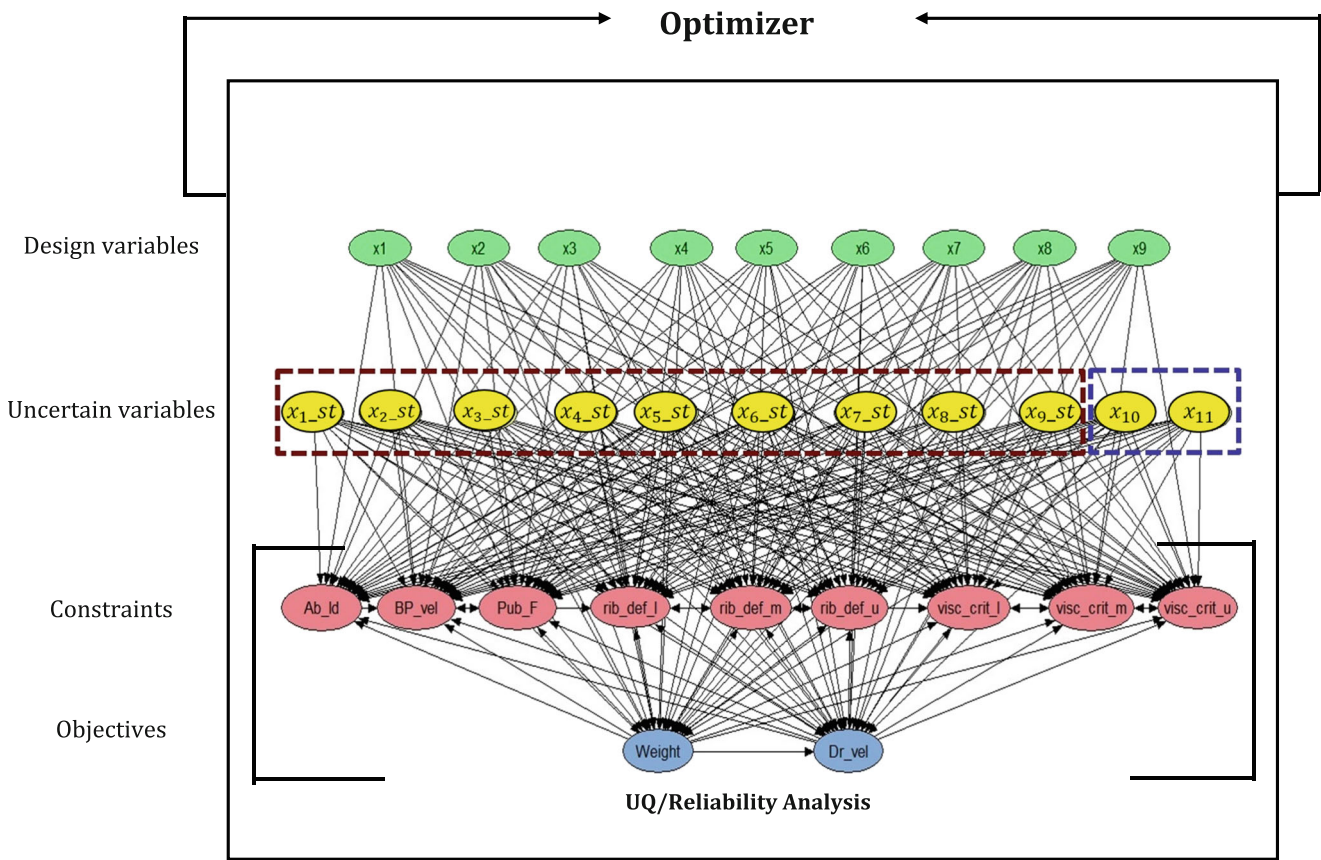


Fig. 7 Optimization with BN

is to compare the BNC method with the SR model and study the extent of agreement between the two approaches.

- Case II: Multi-objective optimization (RBDO formulation) with 2 objectives and 9 individual probability constraints. The objectives are to minimize the mean values of both the car weight, and the velocity of the front door subject to hitting.
- Case III: Multi-objective optimization (RBDO formulation) with 2 objectives, 9 individual probability constraints and an additional joint probability constraint of the 9 outputs. The same objectives as Case II are used, which are minimizing the mean of both the car weight, and the door velocity subject to hitting.

5.1 Case I: single objective RBDO

The problem is formulated as:

$$\begin{aligned}
 & \min_{\mu_x} \mu_{Weight} \\
 s.t. \quad & P_i(Con_i < Crit_i) \geq 0.99, \quad i = 1 \dots 10 \\
 & 0.5 \leq x_j \leq 1.5, \quad j = 1 \dots 9
 \end{aligned} \tag{9}$$

The Dividing RECTangles (DIRECT) algorithm, which is a gradient-free global optimizer, is adopted to solve the optimization problem. The optimum obtained from the BNC approach is compared against the solution of the SR model. The optimization histories is shown in Fig. 8, and the optimal solution is listed in Table 4. The objectives and constraints using the two models are listed in Table 5.

It can be seen from Tables 4 and 5 that the overall performances of the SR and BN approaches are quite similar. The differences between the BN and SR results for objective, constraints and most of the design variables are 5% or less. The probabilistic constraints for pubic symphysis force and door velocity (bold in red) are slightly violated, which may be due to the lack of sufficient samples in training the BN; this issue which will be studied later when discussing Pareto surface improvement. Having verified the accuracy of the BNC approach, the next two cases are implemented for multi-objective RBDO.

5.2 Cases II/III: multi-objective RBDO with individual/joint probability constraints

The optimization with two competing objectives and 9 individual probability constraints (and one additional joint

Table 3 Correlations between output variables

| | Weight | V_{dr} | L_{ab} | F_{pb} | Rib_u | Rib_m | Rib_l | VC_u | VC_m | VC_l | VB_P |
|----------|--------------|--------------|----------|----------|---------|---------|---------|--------|--------|--------|--------|
| Weight | 1.00 | -0.34 | -0.74 | -0.71 | -0.38 | -0.54 | -0.67 | -0.58 | -0.37 | -0.67 | -0.62 |
| V_{dr} | -0.34 | 1.00 | 0.07 | 0.13 | -0.14 | 0.02 | 0.01 | -0.17 | -0.24 | 0.13 | 0.01 |
| L_{ab} | -0.74 | 0.07 | 1.00 | 0.86 | 0.35 | 0.49 | 0.64 | 0.35 | 0.26 | 0.67 | 0.60 |
| F_{pb} | -0.71 | 0.13 | 0.86 | 1.00 | 0.04 | 0.16 | 0.31 | 0.16 | 0.01 | 0.40 | 0.27 |
| Rib_u | -0.38 | -0.14 | 0.35 | 0.04 | 1.00 | 0.95 | 0.80 | 0.85 | 0.94 | 0.56 | 0.87 |
| Rib_m | -0.54 | 0.02 | 0.49 | 0.16 | 0.95 | 1.00 | 0.93 | 0.84 | 0.85 | 0.74 | 0.96 |
| Rib_l | -0.67 | 0.01 | 0.64 | 0.31 | 0.80 | 0.93 | 1.00 | 0.77 | 0.72 | 0.87 | 0.96 |
| VC_u | -0.58 | -0.17 | 0.35 | 0.16 | 0.85 | 0.84 | 0.77 | 1.00 | 0.92 | 0.53 | 0.79 |
| VC_m | -0.37 | -0.24 | 0.26 | 0.01 | 0.94 | 0.85 | 0.72 | 0.92 | 1.00 | 0.43 | 0.78 |
| VC_l | -0.67 | 0.13 | 0.67 | 0.40 | 0.56 | 0.74 | 0.87 | 0.53 | 0.43 | 1.00 | 0.77 |
| VB_P | -0.62 | 0.01 | 0.60 | 0.27 | 0.87 | 0.96 | 0.96 | 0.79 | 0.78 | 0.77 | 1.00 |

probability constraint for case III) is formulated as shown in (10).

$$\begin{aligned}
 & \min_{\mu_x} \quad \mu_{Weight} \ \& \ \mu_{Vel_{door}} \\
 s. \ t. \quad & P_i(Con_i < Crit_i) \geq 0.99 \\
 & P\left(\bigcap_{i=1}^9 (Con_i < Crit_i)\right) \geq 0.99 \text{ (case III only)} \quad (10) \\
 & 0.5 \leq x_j \leq 1.5, \quad j = 1 \dots 7 \\
 & 0.192 \leq x_k \leq 0.345, \quad k = 8, 9
 \end{aligned}$$

The NSGA-II algorithm is applied to construct the Pareto front. The population size in this genetic algorithm implementation is chosen as 150, probability of crossover is 1 and probability of mutation is 0.15. The total number of iterations is 20. MOO with the SR model is first implemented to identify the effect of the additional joint probability constraint.

It can be concluded from Fig. 9 that, with the consideration of the joint probability constraint, the designs along the Pareto front with joint probability are generally above those without the joint probability constraint, especially in the region of low weight and high velocity. This is intuitively correct since the objectives are minimized; with an additional constraint, both objectives have higher values.

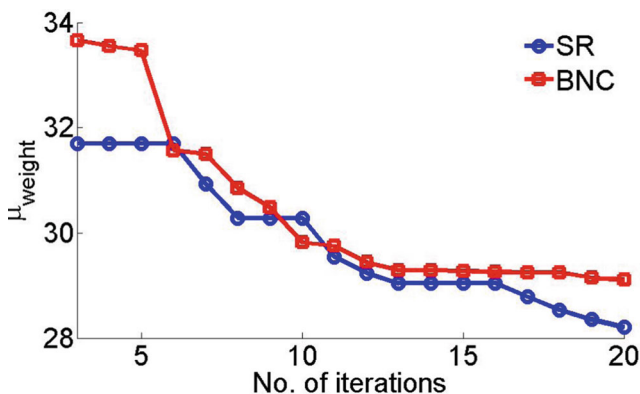


Fig. 8 Single objective RBDO histories for SR and BNC

The proposed BNC approach is next used to construct the Pareto fronts for Cases II and III. At each point of the BNC Pareto front, the objectives are re-evaluated using the SR model. Figure 10 compares the BNC solutions (circles); the BNC solutions re-evaluated using SR (squares) and the SR solutions (triangles) with and without the joint probability constraint. 40 samples are used to depict the Pareto front.

It can be observed from both figures that the BNC Pareto surfaces are more optimistic compared to the SR solution, whereas the re-evaluated results using SR (square marks) are more conservative. The figures also show that the initial BNC approach cannot identify solutions at the high weight (greater than 30), low velocity (less than 14) region.

To investigate this issue, scatter plots that characterize the dependence of weight and door velocity samples from the initial 115 training samples (left) and the samples generated from the Gaussian copula (right) are shown in Fig. 11. The dashed squares cover the region which BNC could not detect in Fig. 10 (i.e., high μ_{weight} , low $\mu_{velocity}$). It can be observed from Fig. 11a that only 3 of the initial samples land within the dashed area. The Bayesian network will only generate samples based on the training points available; in other words, the training points indicate the joint probability distribution of the variables, and therefore further samples generated by the Bayesian network also reflect this joint distribution. Thus the Bayesian network generates very few samples in the dashed area in Fig. 11b and is unable to find Pareto solutions in this area.

To overcome this issue, the training point selection technique proposed in Section 3.4 is applied: 20 additional values of the design variables are generated selectively based on the sculpting strategy. To evaluate this method, Latin Hypercube sampling (LHS) is also used to generate 20 random values of the design variables from uniform distributions based on Tables 1 and 2, and propagated to estimate the outputs. The values of the objectives generated by the two approaches are compared by the scatter plot in Fig. 12.

Table 4 Comparison of optimal solutions using SR and BNC

| | Design variables | RBDO with SR | RBDO with BNC | Difference (%) |
|------------------|--------------------------|--------------|---------------|----------------|
| Design variables | B-pillar inner | 0.96 | 1 | 4 |
| | B-pillar reinforce | 1.18 | 1.33 | 13 |
| | Floor side inner | 0.63 | 0.65 | 3 |
| | Cross member | 1.30 | 1.33 | 2 |
| | Door beam | 0.89 | 0.85 | 4 |
| | Door belt line | 1.44 | 0.96 | 33 |
| | Roof rail | 0.89 | 0.84 | 6 |
| | Mat. of B-pillar inner | 0.34 | 0.32 | 6 |
| | Mat. of floor side inner | 0.27 | 0.32 | 19 |

It can be seen from Fig. 12 that, the selectively generated training points (squares) focus more on the desired/rectangular region compared with the LHS approach. The additional 20 training points generated in different methods combined with the 115 initial training points are used to build new BN models for Case III (with the joint probability constraint). The objectives are re-evaluated using the SR model. All results are compared in Fig. 13.

In Fig. 13a, the LHS-based solution does not have much improvement compared to Fig. 13b. The low door velocity, high weight region is still not covered by the additional samples that are uniformly generated. However, in Fig. 13b, due to the selective resampling based on ‘sculpting’, the improved BNC is able to construct the Pareto surface in the high weight and low door velocity region which could not be reached by the original BNC. And the resulting Pareto surface is very close to the SR solution. This shows that the sculpting strategy can be effectively used to improve the Bayesian network model and the Pareto surface.

Once again, it is worth repeating that here the SR model is treated as the original model, and the BN is the surrogate model. In practical problems, the original model will be expensive, and the BN surrogate model will be trained with selected evaluations of the original model. The methodology proposed here addresses both efficiency (training points) and accuracy (Pareto surface) of the resulting optimal solutions.

6 Conclusion

This paper proposed a novel Bayesian network-based approach for multi-objective optimization under uncertainty. The Bayesian network is constructed as a probabilistic surrogate model based on input–output samples from the original model. A vine copula-based sampling technique is used for efficient uncertainty propagation. A vehicle side impact problem is used to demonstrate the proposed methodology. The joint probability of multiple constraints and objectives are efficiently estimated using the proposed BNC approach, by

Table 5 Performance comparison of the optimal solutions using SR and BNC

| | Output variables (a) | RBDO with SR (b) | RBDO with BNC (c) | Evaluation of BNC with SR (d) | Difference between (b) and (d) % |
|---------------------------|-------------------------|------------------|-------------------|-------------------------------|----------------------------------|
| Objective | Weight | 28.20 | 29.11 | 29.49 | 5 |
| Probabilistic Constraints | Abdomen load | 1 | 1 | 1 | 0 |
| | Upper rib deflection | 1 | 0.99 | 1 | 0 |
| | Middle rib deflection | 1 | 1 | 1 | 0 |
| | Lower rib deflection | 0.99 | 1 | 1 | 1 |
| | Upper viscous criteria | 1 | 0.99 | 1 | 0 |
| | Middle viscous criteria | 1 | 1 | 1 | 0 |
| | Lower viscous criteria | 1 | 1 | 1 | 0 |
| | Pubic force | 0.99 | 1 | 0.97 | 2 |
| | B-pillar velocity | 1 | 1 | 1 | 0 |
| | Door velocity | 1 | 0.99 | 0.97 | 3 |

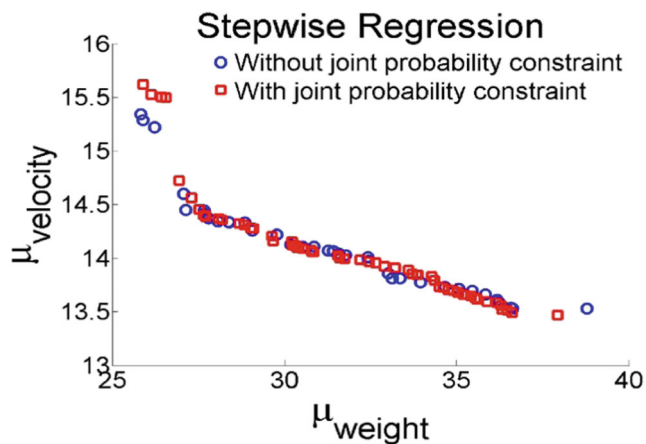


Fig. 9 Comparison of Pareto fronts with/without joint probability constraints

exploiting the forward propagation capability of the Bayesian network.

A novel training point selection technique is proposed to refine the BN model and improve the Pareto surface. Additional samples of training points are generated in the desired region based on sculpting, which exploits the dependence relations among the inputs and outputs, and the inverse propagation capability of the Bayesian network through conditional sampling.

Future work may study the use of analytical multi-normal integration of the Gaussian copulas instead of the sample-based strategy used here to compute the constraint probabilities; this could further improve the efficiency of reliability assessment and optimization. Efficiency improvements in the presence of non-Gaussian copula also need to be studied, since sampling with non-Gaussian copula is much more time-consuming. However, note that the computational effort of the physics model is much larger than that for sampling, even if non-Gaussian copulas are used. The numerical example had 9 design variables, 11 uncertain variables, 9 constraints and 2 objectives; further studies may investigate the performance of the proposed methodology for much larger multi-objective optimization problems. Also, the uncertainty sources

considered in the framework currently only include input variability and data uncertainty. Future studied should be engaged to include model errors in such framework could be studied in the future.

Acknowledgements The Uninet Bayesian network software developed by Lightwist Software is used to build the Bayesian network and perform conditional sampling in this paper. The VisualDOC software by Vanderplaats R&D is used as the optimizer, which provided the NSGA-II algorithm for the multi-objective optimization. The valuable discussions with Prof. Roger Cooke (TU Delft), Prof. Anca Hanea (University of Melbourne), Dan Ababei (Lightwist Software) and Dr. Hong Dong (Vanderplaats R&D, Inc.) are deeply appreciated.

Appendix

Vine representation of BN

The vine approach is a way to identify a set of conditional bivariate joint distributions that represent the joint distribution of all the variables in the problem. Detailed theory on vines can be found in (Cooke and Kurowicka 2007; Jones et al. 1998; Cooke et al. 2015; Youn et al. 2004; Yule and Kendall 1965). Consider the model shown in Fig. 14, and assume that paired samples of X_1, X_2, Y_1 and Y_2 are available. A vine structure can be constructed as shown in Fig. 15, where r_{ij} represent the rank correlations and conditional rank correlations between two variables. The rank correlations represented by the 6 edges in the BN in Fig. 14b are mapped to three non-partial bivariate correlations (solid lines) and three conditional bivariate correlations (dashed lines) in the vine structure. It has been proved that these 6 correlations along with the marginal distributions are able to uniquely define the joint distribution of all the variables in the BN in Fig. 14b (Hanea 2010).

Since the vine approach uniquely represents the multivariate joint distribution using the marginal distributions of the variables and the six correlation terms shown in Fig. 15, only a *bivariate copula* function C needs to be assumed for each of

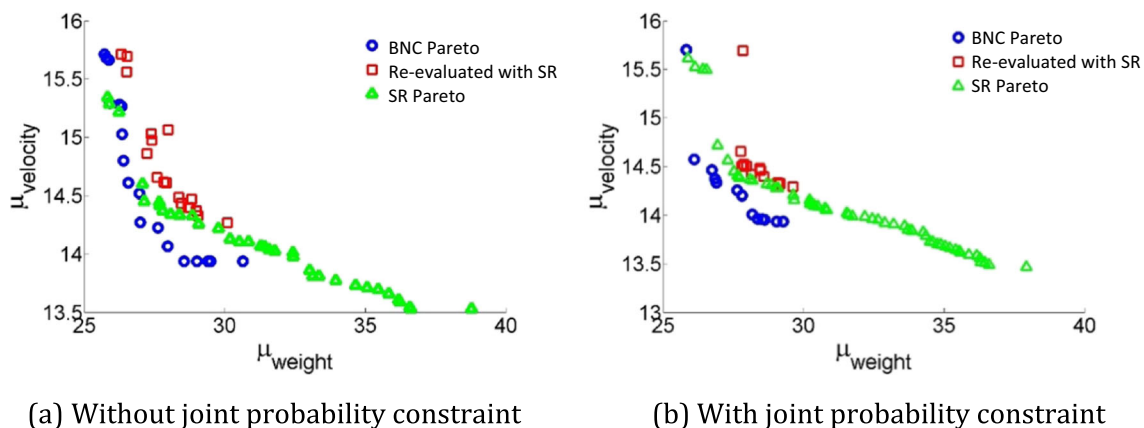
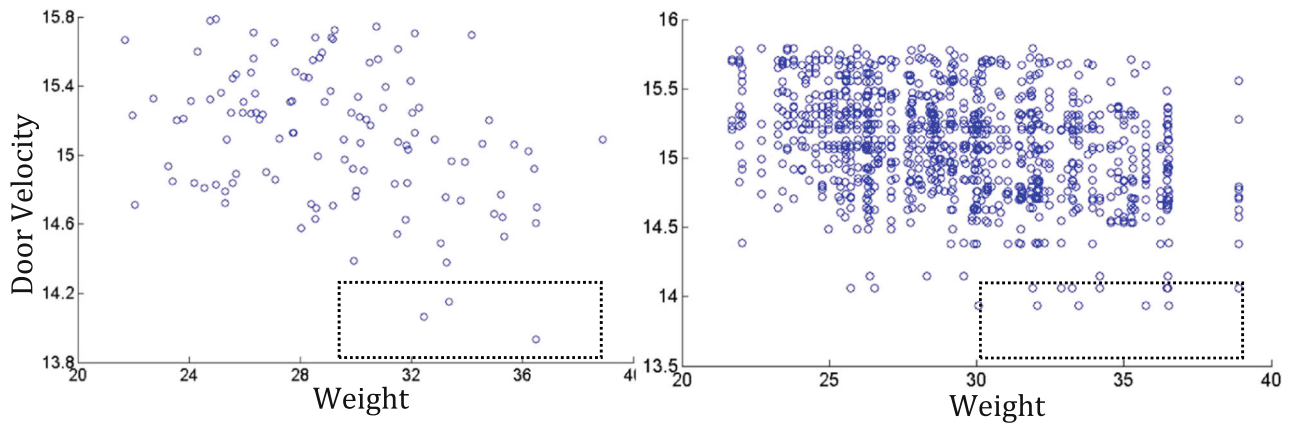


Fig. 10 Pareto fronts with BNC, and SR solutions



(a) Scatter plots from the original samples (b) Scatter plots from the copula-generated samples

Fig. 11 Scatter plots of weight and door velocity

the bivariate distributions. Note that the choice of copula can be different for different edges in a vine structure. Now, suppose we generate samples of 4 independent uniform random variables u_1, u_2, u_3 and u_4 from the interval $[0,1]$. The CDF values of the four variables in the BN, which are correlated, can be obtained using (A.1) (Kurowicka and Cooke 2010).

$$u_{x_1} = u_1 \tag{A.1.1}$$

$$u_{x_2} \Big|_{u_{x_1}} = C^{-1}_{U_{X_2} U_{X_1}} \Big|_{u_{x_1}} (u_2) \tag{A.1.2}$$

$$u_{y_1} \Big|_{u_{x_2} u_{x_1}} = C^{-1}_{U_{Y_1} U_{X_2}} \Big|_{u_{x_2}} \left(C^{-1}_{U_{Y_1} U_{X_1}} \Big|_{u_{x_1} u_{x_2}} (u_3) \right) \tag{A.1.3}$$

$$u_{y_2} \Big|_{u_{y_1} u_{x_2} u_{x_1}} = C^{-1}_{U_{Y_2} U_{Y_1}} \Big|_{u_{y_1}} \left\{ C^{-1}_{U_{Y_2} U_{X_2}} \Big|_{u_{y_1} u_{x_2}} \left[C^{-1}_{U_{Y_2} U_{X_1}} \Big|_{u_{y_1} u_{x_2}} (u_4) \right] \right\} \tag{A.1.4}$$

where C are bivariate copula functions between the CDF of two variables. $C^{-1}_{ij|j}$ indicates the inverse CDF of the

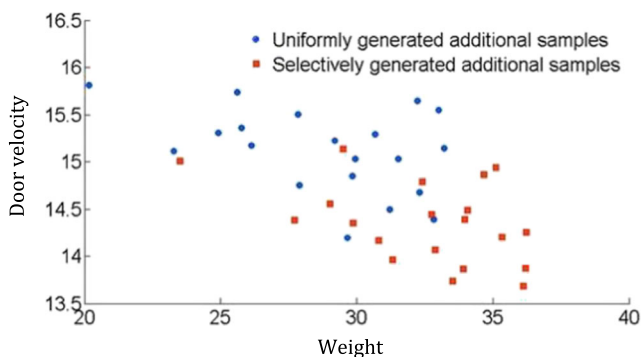


Fig. 12 Comparison of the outputs from different resampling approaches

conditional probability F_{ij}^{-1} . That is, given the CDF(s) of j^{th} variable(s), the inverse CDF with respect to u_i is taken to compute the conditional CDF of i^{th} variable. For the sake of illustration, consider the copula function between X_1 and X_2 :

$$C_{U_{X_1} U_{X_2}} = P(u_{X_1} \leq u_1, u_{X_2} \leq u_2) \tag{A.2}$$

Given a realization of u_2 , and the value of u_{X_1} already generated in (A.1.1), the CDF of X_2 can be calculated as

$$u_{X_2} = C^{-1}_{U_{X_1} U_{X_2}} \Big|_{u_{X_1}} (u_2) = F^{-1}_{U_{X_2}} \Big|_{u_{X_1}} (u_2) \tag{A.3}$$

Similarly, the CDF values of Y_1 and Y_2 , namely u_{Y_1} and u_{Y_2} , can be generated using the bi-variate copulas in (A.1.3) and (A.1.4). Once the *correlated* CDF values are generated, the inverse CDF estimation is subsequently used to obtain the samples of the corresponding variables, as shown in (A.4).

$$x_1 = F^{-1}_{X_1}(u_{x_1}) \tag{A.4.1}$$

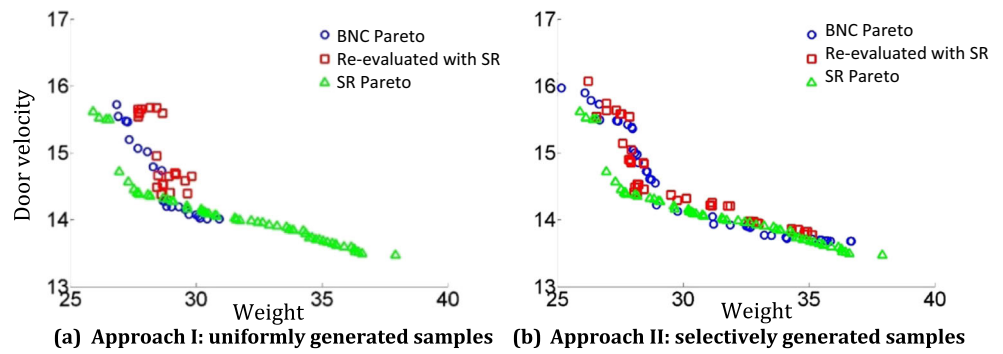
$$x_2 = F^{-1}_{X_2} \left(u_{x_2} \Big|_{u_{x_1}} \right) \tag{A.4.2}$$

$$y_1 = F^{-1}_{Y_1} \left(u_{y_1} \Big|_{u_{x_2} u_{x_1}} \right) \tag{A.4.3}$$

$$y_2 = F^{-1}_{Y_2} \left(u_{y_2} \Big|_{u_{y_1} u_{x_2} u_{x_1}} \right) \tag{A.4.4}$$

When the number of variables within the BN (hence the vine structure) is large, this series of inverse copula estimations can be computationally intensive. The assumption of a Gaussian copula provides an analytical solution and avoids the sequential bivariate estimation, thus making the sampling

Fig. 13 Comparison of the results between the two resampling approach **a** Approach I: uniformly generated samples **b** Approach II: selectively generated samples



inexpensive. A Gaussian copula represents the joint CDF of all marginal CDFs using a multivariate Gaussian distribution.

To apply the Gaussian copula, the Spearman’s rank correlations between all the pairs of variables are first computed, denoted as r_{ij} . Then, Pearson’s transform (Kurowicka and Cooke 2010) is applied to get the conditional linear product moment correlations:

$$\rho_{ij} = 2\sin\left(\frac{r_{ij}\pi}{6}\right) \tag{A.5}$$

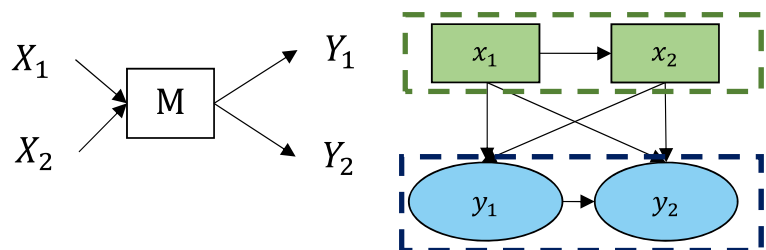
As mentioned earlier, the vine structure is a saturated graph, which is not necessarily true for BN. A missing link in the BN can be expressed in a vine structure by setting the corresponding rank correlation (conditional or unconditional) as zero. Subsequently, the conditional product moment ρ also equal to zero. Then, the bivariate unconditional product moment can be recalculated with the recursive formula (Yule and Kendall 1965) (for a Gaussian copula) as:

$$\rho_{12;3\dots n} = \frac{\rho_{12;3\dots,n-1} - \rho_{1n;3\dots,n-1} \cdot \rho_{2n;3\dots,n-1}}{\sqrt{1 - \rho_{1n;3\dots,n-1}^2} \sqrt{1 - \rho_{2n;3\dots,n-1}^2}} \tag{A.5}$$

A Gaussian copula that represents the relationships in Fig. 15 can be written as:

$$c_R(\mathbf{u}) = \frac{1}{\sqrt{\det(\mathbf{R})}} \exp\left(-\frac{1}{2} \begin{pmatrix} \Phi^{-1}(u_1) \\ \Phi^{-1}(u_2) \\ \Phi^{-1}(u_3) \\ \Phi^{-1}(u_4) \end{pmatrix} \cdot (\mathbf{R}^{-1} - \mathbf{I}) \cdot \begin{pmatrix} \Phi^{-1}(u_1) \\ \Phi^{-1}(u_2) \\ \Phi^{-1}(u_3) \\ \Phi^{-1}(u_4) \end{pmatrix}\right) \tag{A.6}$$

Fig. 14 Bayesian network representation of a model with input and output variables



(a) Two-input-two-output model M

(b) BN representation

where Φ^{-1} represents the inverse CDF of a standard normal random variable. \mathbf{u} are independent uniform random variables from the interval $[0,1]$. \mathbf{I} is an identity matrix. \mathbf{R} is the covariance matrix of the four variables, and since the marginals of the Gaussian copula are standard normals, \mathbf{R} is essentially the correlation coefficient matrix composed of the unconditional product moment correlations ρ_{ij} .

The multivariate Gaussian distribution in (A.6) can be used to rapidly generate a large number of samples of correlated normal random variables. In this case, samples of 4 variables from this joint normal distribution are generated and denoted as X'_1, X'_2, Y'_1, Y'_2 . For each sample of the variables, compute the CDF with respect to the marginal distributions of standard normal distribution, and denote the CDF values as $u_{X_1}, u_{X_2}, u_{Y_1}$ and u_{Y_2} . Samples of X_1, X_2, Y_1 and Y_2 are then obtained by taking the inverse CDFs of $u_{X_1}, u_{X_2}, u_{Y_1}$ and u_{Y_2} with respect to their marginal distributions as shown in (A.4).

Conditional sampling

The combination of Bayesian network and vine copula-based sampling technique (BNC) helps to formulate a methodology for efficient modeling and sampling of a multivariate joint distribution. To use this framework as a surrogate model, the conditional distributions of outputs \mathbf{y} for given values of input \mathbf{x} needs to be estimated. Conditionally sampling with the Gaussian copula assumption is very easy to implement since (A.4) can be converted to conditional Gaussian copula analytically. The procedure is as follows:

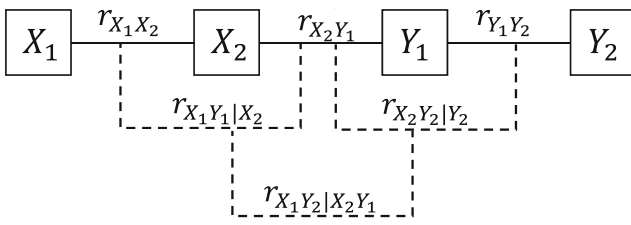


Fig. 15 Vine representation of the BN shown in Fig. 14

For example, the conditional samples of Y_1 and Y_2 need to be generated given $X_1 = x_1, X_2 = x_2$. The equivalent normals corresponding to $X_1 = x_1, X_2 = x_2$ are first calculated as $x'_1 = \Phi^{-1}(F_{X_1}(x_1)), x'_2 = \Phi^{-1}(F_{X_2}(x_2))$.

Let μ be the mean vector of X_1, X_2, Y_1 and Y_2 in the equivalent normal space $[X'_1, X'_2, Y'_1, Y'_2]$; μ is a vector of zeros with 4 entries, and R is the covariance matrix:

$$R = \left[\begin{array}{cc|cc} \Sigma_3 & & \Sigma_1 & \\ \hline 1 & \rho_{12} & \rho_{13} & \rho_{14} \\ & 1 & \rho_{23} & \rho_{24} \\ \hline & & 1 & \rho_{34} \\ & & & 1 \end{array} \right] \Sigma_2$$

Then the conditional joint distribution of Y'_1 and Y'_2 given $X'_1 = x'_1, X'_2 = x'_2$ is denoted as: $f(Y'_1, Y'_2 | X'_1 = x'_1, X'_2 = x'_2) \sim N(\tilde{\mu}, \tilde{\Sigma})$, where the conditioned mean vector $\tilde{\mu}$ and covariance matrix $\tilde{\Sigma}$ are

$$\tilde{\mu} = \Sigma_1 \Sigma_3^{-1} * \begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} \tag{A.7.1}$$

$$\tilde{\Sigma} = \Sigma_2 - \Sigma_1 \Sigma_3^{-1} \Sigma_1^T \tag{A.7.2}$$

Samples Y'_1 and Y'_2 are jointly generated from a multivariate normal distribution, of which the mean and covariance matrix are calculated as in (A.7). The CDF values of each Y'_1 and Y'_2 sample with respect to the standard normal distribution (u_{Y_1}, u_{Y_2}) are computed, and the inverse CDF is taken to obtain the conditional samples of Y_1 and Y_2 as shown in (A.4). Thus sampling from the Gaussian copula avoids the evaluation of the series of inverse copula functions in (A.1), and is very efficient.

References

Bedford T, Cooke RM (2001) Probability density decomposition for conditionally dependent random variables modeled by vines. *Ann Math Artif Intell* 32:245–268

Casella G, George E (1992) Explaining the Gibbs sampler. *Am Stat* 46: 167–174

Chib S, & Greenberg E (1995) Understanding the metropolis-hastings algorithm. *The American Statistician* 49(4):327–335

Choi K, Noh Y, Lee I (2010) Reliability-based design optimization with confidence level for problems with correlated input distributions, 6th China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems

Cooke RM (1997) Markov and entropy properties of tree-and vine-dependent variables., *Proceedings of the ASA Section of Bayesian Statistical Science*

Cooke R, Kurowicka D (2007) Continuous/discrete non-parametric bayesian belief nets with unicorn and uninet, *Proceedings of Mathematical Methods in Reliability MMR*

Cooke RM, Zang TA, Mavris DN, JCT (2015) Sculpting: a fast, interactive method for probabilistic design space exploration and margin allocation, 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6:182–197

Du L, Choi KK, Youn BD, Gorsich D (2006) Possibility-based design optimization method for design problems with both statistical and fuzzy input data. *J Mech Des* 128:928–935

Ferson S, Kreinovich V, Hajagos J, Oberkampf W, Ginzburg L (2007) Experimental uncertainty estimation and statistics for data having interval uncertainty, Sandia national laboratories technical report. Albuquerque, New Mexico

Gollwitzer S, Rackwitz R (1988) An efficient numerical solution to the multinormal integral. *Probab Eng Mech* 3:98–101

Guo J, Du X (2009) Reliability sensitivity analysis with random and interval variables. *Int J Numer Methods Eng* 78:1585–1617

Haldar A, Mahadevan S (2000) Probability, reliability, and statistical methods in engineering design. John Wiley

Hanea A (2010) Non-parametric bayesian belief nets versus vines, *Dependence modeling: vine copula handbook*, 2010, p. 281

Hohenbichler M, Rackwitz R (1983) First-order concepts in system reliability. *Struct Saf* 1:177–188

Jiang X, Mahadevan S (2008) Bayesian probabilistic inference for non-parametric damage detection of structures. *J Eng Mech* 134:820–831

Jiang X, Mahadevan S (2009) Bayesian hierarchical uncertainty quantification by structural equation modeling. *Int J Numer Methods Eng* 80:717–737

Jones D, Schonlau M, Welch W (1998) Efficient global optimization of expensive black-box functions. *J Glob Optim* 13:455–492

Knotters M, Brus DJ, Oude Voshaar JH (1995) A comparison of kriging, co-kriging and kriging combined with regression for spatial interpolation of horizon depth with censored observations. *Geoderma* 67: 227–246

Kurowicka D, Cooke RM (2010) Vines and continuous non-parametric bayesian belief nets with emphasis on model learning, *Rethinking Risk Measurement and Reporting I*, pp. 295–330

Kurowicka D, & Cooke RM (2006) Uncertainty analysis with high dimensional dependence modelling. John Wiley & Sons.

Liang C, Mahadevan S (2013) Stochastic multidisciplinary analysis with high-dimensional coupling, 10th World Congress on Structural and Multidisciplinary Optimization

Liang C, Mahadevan S (2015) Bayesian sensitivity analysis and uncertainty integration for robust optimization. *J Aerosp Inform Syst* 12: 1–42

Ling Y, Mullins J, Mahadevan S (2014) Selection of model discrepancy priors in Bayesian calibration. *J Comput Phys* 276:665–680

Liu Y, Chen W, Arendt P, Huang H-Z (2011) Toward a better understanding of model validation metrics. *J Mech Des* 133:071005

- Mahadevan S, Liang B (2011) Error and uncertainty quantification and sensitivity analysis in mechanics computational models. *Int J Uncertain Quantif* 1:147–161
- Mahadevan S, Rebba R (2005) Validation of reliability computational models using bayes networks. *Reliab Eng Syst Saf* 87:223–232
- Marler RT, Arora JS (2004) Survey of multi-objective optimization methods for engineering. *Struct Multidiscip Optim* 26:369–395
- Mavrotas G (2009) Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems. *Appl Math Comput* 213:455–465
- Neal RM (2003) Slice sampling. *Ann Stat* 31:705–741
- Nelsen RB (1999) *An introduction to copulas*, Springer
- Noh Y, Choi KK, Du L (2008) Reliability-based design optimization of problems with correlated input variables using a Gaussian copula. *Struct Multidiscip Optim* 38:1–16
- Rangavajhala S, Mahadevan S (2011) Joint probability formulation for multiobjective optimization under uncertainty. *J Mech Des* 133: 051007
- Rebba R, Mahadevan S, Huang S (2006) Validation and error estimation of computational models. *Reliab Eng Syst Saf* 91:1390–1397
- Riley ME, Grandhi RV (2011) Quantification of model-form and predictive uncertainty for multi-physics simulation. *Comput Struct* 89: 1206–1213
- Sankararaman S (2012) *Uncertainty quantification and integration*, Vanderbilt University
- Sankararaman S, Mahadevan S (2011) Likelihood-based representation of epistemic uncertainty due to sparse point data and/or interval data. *Reliab Eng Syst Saf* 96:814–824
- Smith N, Mahadevan S (2005) Integrating system-level and component-level designs under uncertainty. *J Spacecr Rocket* 42
- Valdebenito MA, Schuëller GI (2010) A survey on approaches for reliability-based optimization. *Struct Multidiscip Optim* 42:645–663
- Youn BD, Choi KK, Yang RJ, Gu L (2004) Reliability-based design optimization for crashworthiness of vehicle side impact. *Struct Multidiscip Optim* 26:272–283
- Yule GU, Kendall MG (1965) *An introduction to the theory of statistics*. Charles Griffin & Co, Belmont, CA
- Zaman K, Rangavajhala S, McDonald MP, Mahadevan S (2011) A probabilistic approach for representation of interval uncertainty. *Reliab Eng Syst Saf* 96:117–130
- Zhang X, Huang H-Z (2009) Sequential optimization and reliability assessment for multidisciplinary design optimization under aleatory and epistemic uncertainties. *Struct Multidiscip Optim* 40:165–175
- Zhang R, Mahadevan S (2000) Model uncertainty and bayesian updating in reliability-based inspection. *Struct Saf* 22:5–9
- Zhang Y, Monder D, Fraser Forbes J (2002) Real-time optimization under parametric uncertainty: a probability constrained approach. *J Process Control* 12:373–389