SIAM/ASA J. UNCERTAINTY QUANTIFICATION
Vol. 4, pp. 1060–1083



# Shapley Effects for Global Sensitivity Analysis: Theory and Computation*

Eunhye Song†, Barry L. Nelson†, and Jeremy Staum†

**Abstract.** Variance-based global sensitivity analysis decomposes the variance of the output of a computer model, resulting from uncertainty about the model's inputs, into variance components associated with each input's contribution. The two most common variance-based sensitivity measures, the first-order effects and the total effects, may fail to sum to the total variance. They are often used together in sensitivity analysis, because neither of them adequately deals with interactions in the way the inputs affect the output. Therefore Owen proposed an alternative sensitivity measure, based on the concept of the Shapley value in game theory, and showed it always sums to the correct total variance if inputs are independent. We analyze Owen's measure, which we call the Shapley effect, in the case of dependent inputs. We show empirically how the first-order and total effects, even when used together, may fail to appropriately measure how sensitive the output is to uncertainty in the inputs when there is probabilistic dependence or structural interaction among the inputs. Because they involve all subsets of the inputs, Shapley effects could be expensive to compute if the number of inputs is large. We propose a Monte Carlo algorithm that makes accurate approximation of Shapley effects computationally affordable, and we discuss efficient allocation of the computation budget in this algorithm.

**Key words.** Shapley value, computer experiments, global sensitivity

**AMS subject classifications.** 62K99, 62P30, 68U20

**DOI.** 10.1137/15M1048070

## 1. Introduction.

A computer model has a collection of rules to map its inputs, which represent the features of the target system, to one or more outputs. Often, the inputs of the model are uncertain, which means that the resulting output can be regarded as random. This uncertainty in the inputs can be due to lack of information on the exact value of an input, e.g., the recovery rate from a disease (Saltelli et al. [22]), or due to the random nature of an environmental factor that is an input, e.g., the wind speed at the site of a forest fire (Salvador et al. [25]). In the former case, we may reduce the uncertainty in the input by investing more resources, e.g., collecting real-world observations or performing a scientific investigation. In the latter case, the uncertainty is inherent.

Sensitivity analysis studies the fluctuation in a model output caused by changing the model inputs. Local sensitivity analysis focuses on the sensitivity of the output to perturbing the input around a particular value. However, as Saltelli et al. [22] point out, when the input is random, it is difficult to justify local sensitivity analysis not knowing which value the input

*Received by the editors November 12, 2015; accepted for publication (in revised form) July 18, 2016; published electronically September 7, 2016.

http://www.siam.org/journals/juq/4/M104807.html

**Funding:** This research was partially supported by National Science Foundation grant CMMI-1068473 and GOALI sponsor Simio LLC.

†IEMS, Northwestern University, Evanston, IL 60208 (eunhyesong2016@u.northwestern.edu, nelsonb@northwestern.edu, j-staum@northwestern.edu).

will take. Global sensitivity analysis, on the other hand, measures the uncertainty in the output caused by the uncertainty in the inputs over the range of possible input values. In this paper, we consider global sensitivity analysis. Different global sensitivity analysis methods are relevant for different purposes. Variance-based sensitivity measures serve many of these purposes by decomposing the variance in the model output and allocating it to each input in the model. Two of the most widely used measures are first-order and total effects suggested by Homma and Saltelli [12]. A first-order effect measures the expected reduction in variance of the output when an input is fixed to a constant, whereas the total effect measures the expected remaining variance of the output when all other input values are fixed.

One purpose of global sensitivity analysis is to provide guidance for investing resources to mitigate the uncertainty in a model output by reducing the uncertainty in the inputs: that is, to decide which input (or set of inputs) to control or to determine more accurately to reduce the variance of the output the most. Saltelli et al. [24] refer to this problem as *factor prioritization*. The first-order effects are defined in a way that makes them useful for factor prioritization.

Global sensitivity analysis can also be used to create a parsimonious model that includes only a small fraction of the inputs to which the output is the most sensitive, which has long been discussed in *factor screening* (Kleijnen [15]). This is particularly critical when we have a data-driven model such that the number of inputs into the model is much larger than the number of observations of the inputs. Breiman [3] uses a *random forest* for factor screening to exclude inputs to which the model has little sensitivity. As each tree in the random forest is built from a bootstrap sample, we have the corresponding out-of-bag (oob) observations that can be used to test each tree. The importance measure of each input is the difference in the number of misspecifications when the observed values of the input (e.g., genders of participants in a clinical study) in the oob samples are shuffled randomly from that of the original samples; i.e., if the model is not sensitive to the input at all, the shuffled oob samples will provide predictions just as good as the original oob samples. Hence, inputs with low importance can be dropped from the model.

However, when the model is derived from an underlying physical law of the system it is difficult to eliminate inputs from the model. An alternative to eliminating an input is to fix its value to a constant, i.e., to ignore uncertainty about the input. Saltelli et al. [24] refer to this problem as *factor fixing*. One benefit of factor fixing is that it allows for lower-dimensional, thus cheaper, experiment designs in sensitivity analysis. Variance-based sensitivity measures can be used in factor fixing to identify inputs whose uncertainty can be ignored without substantially understating uncertainty about the output.

When the goal is to design a system that is robust to random inputs, global sensitivity analysis can provide information on how sensitive the output is to the inputs. For instance, an engineer can identify the environmental factor to which the output of an energy plant is the most sensitive using variance-based sensitivity measures. Combined with knowledge of the system, the engineer can then propose an improved design in an attempt to reduce the sensitivity.

Variance-based sensitivity measures can also be useful to manage tensions in a business. In section 5 we provide an example where we perform sensitivity analysis to find the product

type whose order placement rate affects the average job completion time the most. In such a situation, the company may assign especially adept workers to deal with those product types.

Finally, global sensitivity analysis can be applied to understand a scientific phenomenon. Saltelli, Chan, and Scott [21] perform a sensitivity analysis to study the impact of biological and ecological factors on a fish population for better understanding of the ecosystem. Saltelli and Trarantola [23] analyze the impacts of inputs to the "Level E" model, which predicts the radiologic dose to humans over geologic time scales due to the underground migration of radionuclides from a nuclear waste disposal site. In section 5, we present sensitivity analysis of a fire-spread model for Mediterranean shrublands and identify the environmental factors that affect the spread of fire the most.

In the case of factor prioritization, the goal is to identify inputs with the property that reducing their uncertainty reduces the total output variance the most. Hence, there is no reason to expect that the sum of such sensitivity measures will be equal to the total variance. In other settings above, however, it is important to decompose the variance of the model output into variances assigned to each input so that (1) these variances sum to the total variance, and (2) we have appropriate and easily interpreted information on each input's contribution to the total variance. For the latter reason, Saltelli et al. [24] suggest using the total effects instead of the first-order effects in factor fixing, as the first-order effects can miss interaction effects among inputs. However, the total effect can overestimate an input's effect on the overall variance when we have a nonlinear model with independent inputs, i.e., the sum of the total effects is greater than the total variance of the output. As an alternative, Owen [18] suggests a new sensitivity measure based on the concept of the Shapley value in game theory (Shapley [26]) and shows that it always sums to the total variance of the output. However, his analysis is limited to the case with independent inputs.

We extend the analysis of this new measure to the case of dependent inputs, showing that the Shapley values still sum to the total variance of the output. We compare the performance of the Shapley values to that of the first-order and total effects. When the inputs are correlated, we discover (a) the sum of first-order effects may exceed the total variance of the output, and (b) the sum of the total effects may be lower than the total variance of the output. Both findings contradict the general folklore on the two effects: that the first-order effect underestimates and the total effect overestimates the contribution of an input. We argue that neither effect is a good measure to decompose the variance of the output and suggest the Shapley values as an alternative.

Calculation of the Shapley values may demand substantial computational effort. Castro, Gómez, and Cazorla [7] provide an algorithm to estimate Shapley values given a cost function that can be evaluated exactly. We modify their algorithm to achieve better efficiency and to apply in the context of sensitivity analysis where the cost function needs to be estimated by Monte Carlo simulation. We suggest a reasonable allocation of the computation budget between Shapley value and cost function estimation in the algorithm. Our algorithm in R code is available at http://users.iems.northwestern.edu/~nelsonb/ShapleyEffects. We expect our analysis and software to facilitate global sensitivity analysis of systems with dependent inputs especially in problems where correctly allocating the total output variance to each input is important.

The paper is organized in the following way. In section 2, we review traditional variance-based sensitivity measures. Section 3 reviews the concept of Shapley value in game theory and formally defines the *Shapley effect* followed by an analytical comparison to the first-order and total effects. Section 4 provides an efficient algorithm to estimate the Shapley effects. Section 5 illustrates the use of Shapley effects on a simple manufacturing system model and a realistic fire-spread model and compares its performance to the first-order and total effects.

**2. Variance-based sensitivity analysis.** Consider a model that has $k$ inputs denoted by $\mathbf{X}_{\mathcal{K}} = \{X_1, X_2, \ldots, X_k\}$, where $\mathbf{X}_{\mathcal{J}}$ indicates the vector of inputs included in the index set $\mathcal{J} \subseteq \mathcal{K}$, and $\mathcal{K} = \{1, 2, \ldots, k\}$. The uncertainty in $\mathbf{X}_{\mathcal{K}}$ is represented by the joint cumulative distribution $\mathbf{G}_{\mathcal{K}}$. Further, we denote the joint distribution of inputs included in the index set $\mathcal{J}$ as $\mathbf{G}_{\mathcal{J}}$ and the marginal distribution of each $X_i$ as $G_i$. A design or decision variable whose value can be chosen is not a candidate for variance-based global sensitivity analysis. Variance-based global sensitivity analysis puts a distribution on uncertain inputs and views them as random, but design and decision variables are not randomly chosen. Sensitivity to design and decision variables can be measured by other types of sensitivity analysis.

The model response $Y$ is a function of the inputs, i.e., $Y = \eta(\mathbf{X}_{\mathcal{K}})$, and therefore $\eta(\mathbf{X}_{\mathcal{K}})$ is stochastic due to the uncertainty in $\mathbf{X}_{\mathcal{K}}$, although $\eta(\cdot)$ is deterministic. Often, $\eta(\cdot)$ has a complex structure (e.g., computer codes) and does not have a closed-form expression. The overall uncertainty in the output $Y$ caused by $\mathbf{X}_{\mathcal{K}}$ is $\mathrm{Var}[Y]$, where the variance is taken with respect to the joint distribution, $\mathbf{G}_{\mathcal{K}}$. We are interested in variance-based sensitivity measures that quantify how much of $\mathrm{Var}[Y]$ can be attributed to each $X_i$.

The Sobol' indices (Sobol' [27]) were first introduced to measure the sensitivity of the output to *each subset* $\mathcal{J}$ of $\mathcal{K}$ using functional analysis of variance. Under the assumption of independent inputs, $\mathrm{Var}[Y]$ is decomposed as a sum of variance components attributable to each $\mathcal{J}$. Homma and Saltelli [12] define the *first-order* effect $V_i$ of $X_i$ as the Sobol' index attributable to the main effect of $i$:

$$V_i \equiv \mathrm{Var}[\mathrm{E}[Y|X_i]] = \mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|X_i]]. \tag{1}$$

By definition, $V_i$ leaves out the variability of $Y$ caused by interactions of $X_i$ with other inputs. The second expression in (1) can be interpreted as the expected reduction in $\mathrm{Var}[Y]$ when we fix the value of $X_i$ to a constant. To complement the first-order effect, they define the *total* effect $T_i$ of $X_i$ as

$$T_i \equiv \mathrm{Var}[Y] - \mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{-i}]] = \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-i}]], \tag{2}$$

where $\mathbf{X}_{-i} = \mathbf{X}_{\mathcal{K}\setminus\{i\}}$. In words, $T_i$ is the expected remaining variance of $Y$ when the values of inputs in $\mathcal{K}\setminus\{i\}$ are known. If all inputs are independent and $\eta(\mathbf{X}_{\mathcal{K}})$ is perfectly additive, i.e., $\eta(\mathbf{X}_{\mathcal{K}}) = \sum_{i=1}^{k} f_i(X_i)$ for some functions $f_1, f_2, \ldots, f_k$, then $V_i = T_i$. In general, $V_i < T_i$ when inputs are independent, and they are used together to complement each other. Homma and Saltelli [12] show that with independent inputs $T_i$ is the sum of first-order effect $V_i$ and all the interaction effects by $X_i$ with other inputs. Hence, $T_i - V_i$ is a measure of how much $X_i$ is involved in interactions. With independent inputs

$$\sum_{i=1}^{k} V_i \le \mathrm{Var}[Y] \le \sum_{i=1}^{k} T_i, \tag{3}$$

where the equalities hold if the model is perfectly additive.

Although these sensitivity measures based on Sobol' indices are widely accepted in applications, their fundamental assumption of independence among inputs limits the scope of problems to which these measures can be applied. In fact, in section 3.2 we show that the inequalities in (3) no longer hold in the case of dependent inputs; under some dependence structures, the sum of total effects is less than the sum of the first-order effects, which makes it difficult to interpret the two effects.

As an alternative to Sobol' indices, Chastaing, Gamboa, and Prieur [9] propose a sensitivity index that can be used under dependent inputs. Based on earlier work of Hooker [13], they decompose $\eta(\mathbf{X}_\mathcal{K})$ as a sum of hierarchically orthogonal functions of $\mathbf{X}_\mathcal{J}$ for each $\mathcal{J} \subset \mathcal{K}$. For a certain class of density function for $\mathbf{X}_\mathcal{K}$, they show that this decomposition is unique. Using the decomposition, they define an effect for each subset $\mathcal{J}$ of $\mathcal{K}$ that sums to $\mathrm{Var}[Y]$. Therefore, the purpose of their effects is to decompose $\mathrm{Var}[Y]$ and allocate it to *each subset* $\mathcal{J}$, which is different from decomposing $\mathrm{Var}[Y]$ by allocating it to *each input* $X_i$, which is our objective.

In measuring sensitivity for factor fixing and robust system design, or for studying a scientific phenomenon, it is useful to have a single sensitivity measure for each input that decomposes the effect of each input on the output variance, both independent of and in interactions with other inputs regardless of the dependence among inputs. For this purpose, we propose the *Shapley effect*, a modified definition of Owen's Shapley values (Owen [18]).

## 3. Shapley effect.

This section provides the definition of the Shapley effect and compares its performance to that of the first-order and total effects under dependent inputs.

### 3.1. Definition.

In game theory, the *Shapley value* (Shapley [26]) is used to evaluate the "fair share" of a player in a cooperative game. Formally, a *k-player game* with the set of players $\mathcal{K} = \{1, 2, \ldots, k\}$ is defined as a real-valued function that maps a subset of $\mathcal{K}$ to its corresponding cost (or value), i.e., $c$: $\mathbf{2}^\mathcal{K} \to \mathbb{R}$ with $c(\emptyset) = 0$. Hence, $c(\mathcal{J})$ represents the cost that arises when the players in the subset $\mathcal{J}$ of $\mathcal{K}$ participate in the game. The Shapley value of player $i$ with respect to $c(\cdot)$ is defined as

$$v_i = \sum_{\mathcal{J} \subseteq \mathcal{K}\backslash\{i\}} \frac{(k - |\mathcal{J}| - 1)!|\mathcal{J}|!}{k!} \left(c(\mathcal{J} \cup \{i\}) - c(\mathcal{J})\right), \tag{4}$$

where $|\mathcal{J}|$ indicates the size of $\mathcal{J}$. In words, $v_i$ is the incremental cost of including player $i$ in set $\mathcal{J}$ averaged over all sets $\mathcal{J} \subseteq \mathcal{K}\backslash\{i\}$. Notice that the weight for each incremental cost of size-$s$ subset of $\mathcal{K}\backslash\{i\}$ in (4) can be written as $\frac{(k-s-1)!s!}{k!} = \frac{1}{k}\binom{k-1}{s}^{-1}$. Hence, it gives equal weight, $1/k$, to all $k$ possible sizes of subsets ($s = 0, 1, \ldots, k-1$) and equal weight, $\binom{k-1}{s}^{-1}$, to all possible subsets of size $s$. The notion of Shapley value was generalized to the

*semivalue* by Dubey, Neyman, and Weber [11]. The semivalue of player $i$ with respect to $c(\cdot)$ is defined as

$$\sum_{\mathcal{J}\subseteq\mathcal{K}\setminus\{i\}} p_{|\mathcal{J}|}\left(c(\mathcal{J}\cup\{i\})-c(\mathcal{J})\right), \tag{5}$$

where $p_s$ indicates the relative importance of a size-$s$ subset such that

$$\sum_{s=0}^{k-1}\binom{k-1}{s}p_s = 1.$$

Therefore, the Shapley value is a special kind of semivalue where $p_s = \frac{1}{k}\binom{k-1}{s}^{-1}$, which gives equal weight to each of $k$ subset sizes and equal weights among the subsets of the same size. This gives the Shapley value a distinctive feature among possible semivalues as discussed in section 3.2.

In the context of global sensitivity analysis, we can think of the set of players $\mathcal{K}$ as the set of inputs of $\eta(\cdot)$ and define $c(\cdot)$ so that for $\mathcal{J}\subseteq\mathcal{K}$, $c(\mathcal{J})$ measures the variance of $Y$ caused by the uncertainty of the inputs in $\mathcal{J}$. Clearly, we would like $c(\emptyset)=0$ and $c(\mathcal{K})=\mathrm{Var}[Y]$. For instance, Owen [18] chose

$$\tilde{c}(\mathcal{J}) = \mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{\mathcal{J}}]], \tag{6}$$

which satisfies the two conditions above. Similar to the first-order effect, we can rewrite (6) to $\tilde{c}(\mathcal{J}) = \mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{\mathcal{J}}]]$ and interpret it as the expected reduction in the output variance when the values of $\mathbf{X}_{\mathcal{J}}$ are known. Another choice of the cost function that satisfies the conditions is

$$c(\mathcal{J}) = \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-\mathcal{J}}]], \tag{7}$$

where $\mathbf{X}_{-\mathcal{J}} = \mathbf{X}_{\mathcal{K}\setminus\mathcal{J}}$. Similar to the total effect, (7) is interpreted as the expected remaining variance in $Y$ when the values of $\mathbf{X}_{-\mathcal{J}}$ are known. In this case, the incremental cost $c(\mathcal{J}\cup\{i\})-c(\mathcal{J})$ can be interpreted as the expected decrease in the variance of $Y$ if we are given the input value of $X_i$ out of all the unknown inputs in $\mathcal{J}\cup\{i\}$. The proof of the following theorem is provided in Appendix A.

Theorem 1. *The Shapley values defined using cost function $\tilde{c}$ and $c$ are equivalent.*

Therefore, we define the *Shapley effect* of the $i$th input, $Sh_i$, as the Shapley value obtained by applying the cost function $\tilde{c}$ or $c$ to definition (4). In the rest of the paper, we use $Sh_i$ to denote the Shapley effect and use $v_i$ to denote a generic Shapley value. In section 4.1, we propose an algorithm to estimate Shapley effects defined using $c$ instead of $\tilde{c}$. Although both cost functions result in the same Shapley values, their estimators from Monte Carlo simulation are different. As Sun, Apley, and Staum [29] point out, the two-level Monte Carlo simulation estimator of $\tilde{c}(\mathcal{J})$ can be badly biased unless the inner level sample size to estimate the conditional expectation is quite large. In contrast, the estimator of $c(\mathcal{J})$ is unbiased for all sample sizes. Hence, we chose to estimate $c$ in our algorithm rather than $\tilde{c}$. As an alternative, one can apply "one-and-a-half-level simulation," as suggested by Sun, Apley, and Staum [29], to obtain an unbiased estimator of $\tilde{c}$.

**3.2. Comparison to first-order and total effects.** The first-order effect $V_i$ and the total effect $T_i$ can be defined as semivalues using the cost function in (7). For $V_i$, if we choose $p_s = 0$ for $0 \le s \le k-2$ and $p_{k-1} = 1$, then

$$V_i = \mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|X_i]] = \mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-\mathcal{K}\setminus\{i\}}]] = c(\mathcal{K}) - c(\mathcal{K}\setminus\{i\}). \tag{8}$$

Notice that (8) matches definition (5) for this choice of $p_s$. Similarly, $T_i$ can be obtained by choosing $p_0 = 1$ and $p_s = 0$ for $1 \le s \le k-1$:

$$T_i = \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-i}]] = c(\{i\}) - c(\emptyset). \tag{9}$$

Hence, both first-order and total effects fit into the general framework of semivalues. From the relationship in (15) in Appendix A, the first-order and total effects can also be written as semivalues using Owen's cost function $\tilde{c}$: $p_0 = 1$ and $p_s = 0$ for $1 \le s \le k-1$ gives $V_i$; $p_s = 0$ for $0 \le s \le k-2$ and $p_{k-1} = 1$ gives $T_i$.

One outstanding feature of the Shapley value compared to other semivalues is that it is the only semivalue such that the values of individual players sum to the total cost $c(\mathcal{K})$ regardless of the choice of $c$ (Carreras and Giménez [6]). This supports the benefit of the Shapley effect as

$$\sum_{i=1}^{k} Sh_i = \mathrm{Var}[Y]. \tag{10}$$

In fact, any Shapley values defined by a valid choice of $c$ ($c(\emptyset) = 0$ and $c(\mathcal{K}) = \mathrm{Var}[Y]$) satisfy (10) even if there is dependence or structural interactions among the elements of $\mathbf{X}_{\mathcal{K}}$. As mentioned in section 2, $\sum_{i=1}^{k} V_i \le \mathrm{Var}[Y]$ and $\sum_{i=1}^{k} T_i \ge \mathrm{Var}[Y]$ with independent inputs. However, the following theorem applies when we have dependent inputs.

Theorem 2. *There exists a joint distribution $\mathbf{G}_{\mathcal{K}}$ and function $\eta(\cdot)$ such that $\sum_{i=1}^{k} V_i > \mathrm{Var}[Y] > \sum_{i=1}^{k} T_i$.*

The example in the proof of Theorem 2 in Appendix A has positive correlation between $X_1$ and $X_2$, which makes $V_1$ include some part of the impact caused by $X_2$, as fixing $X_1$ reduces the variability in $X_2$ as well. On the other hand, the positive correlation takes a part of the impact caused by $X_1$ from $T_1$, as fixing $X_2$ also reduces the variability in $X_1$. As a result, the effect due to the correlation of $X_1$ and $X_2$ is included in both $V_1$ and $V_2$, whereas it is overlooked by both $T_1$ and $T_2$, which is the opposite of what we expect in an independent input case.

As the number of inputs increases, and the dependence structure of the inputs and the response function get more complicated, it becomes more difficult to decide which inputs contribute the most to $\mathrm{Var}[Y]$ by comparing the first-order and total effects. One might argue that we can still use the first-order and total effects to find which input has the largest impact on the output by comparing their relative magnitudes. However, in section 5, we provide two examples that show the sorted order of the first-order and total effects inputs are different.

**4. Estimating the Shapley effect.** In this section, we propose an algorithm to estimate the Shapley effects and examine its properties. We also discuss generating dependent inputs for the algorithm.

**4.1. Algorithm.** One barrier to using the Shapley value is its computational complexity as all possible subsets of the players need to be considered. One way to enumerate these subsets is to first consider all possible permutations of the players. Given a permutation $\pi$ of all $k$ players, define the set $P_i(\pi)$ as the players that precede player $i$ in $\pi$. For instance, if $k = 5$ and $\pi = \{1, 3, 2, 4, 5\}$, then $P_2(\pi) = \{1, 3\}$. Hence, the incremental cost of including player $i$ in $P_i(\pi)$ is $c(P_i(\pi) \cup \{i\}) - c(P_i(\pi))$. Taking all possible permutations of $k$ players into consideration, we can rewrite the Shapley value $v_i$ in (4) as

$$v_i = \sum_{\pi \in \Pi(\mathcal{K})} \frac{1}{k!} \left( c(P_i(\pi) \cup \{i\}) - c(P_i(\pi)) \right), \tag{11}$$

where $\Pi(\mathcal{K})$ denotes the set of all $k!$ permutations of players in $\mathcal{K}$ and the incremental cost from each permutation is weighted by $1/k!$. To calculate the Shapley value from (11), we essentially need to evaluate $c(\mathcal{J})$ for all $\mathcal{J} \subset \mathcal{K}$, i.e., $2^k - 1$ variance components and $k!$ permutations. Instead of the exact calculation, Castro, Gómez, and Cazorla [7] provide an approximation algorithm, ApproShapley. Our Algorithm 1 not only modifies ApproShapley to increase the efficiency, it also includes the Monte Carlo simulation of the cost function $c$ to estimate the Shapley effects.

In ApproShapley, $m$ permutations $\pi_1, \pi_2, \ldots, \pi_m$ in $\Pi(\mathcal{K})$ are randomly generated and $v_i$ is estimated by

$$\widehat{v}_i = \frac{1}{m} \sum_{\ell=1}^{m} \left( c(P_i(\pi_\ell) \cup \{i\}) - c(P_i(\pi_\ell)) \right), \tag{12}$$

which is shown to converge in probability to $v_i$. In ApproShapley, if permutation $\pi_\ell$ is generated, then the incremental cost $\Delta_i c(\pi_\ell) = c(P_i(\pi_\ell) \cup \{i\}) - c(P_i(\pi_\ell))$ is calculated for each $i$. For instance, if $k = 3$ and $\pi_\ell = \{1, 3, 2\}$, then ApproShapley calculates

$$\begin{aligned}
\Delta_1 c(\pi_\ell) &= c(\{1\}) - c(\emptyset), \\
\Delta_2 c(\pi_\ell) &= c(\{1, 3, 2\}) - c(\{1, 3\}), \\
\Delta_3 c(\pi_\ell) &= c(\{1, 3\}) - c(\{1\}),
\end{aligned}$$

so $c(\{1, 3\})$ and $c(\{1\})$ are calculated twice. To reduce these redundant calculations our Algorithm 1 below calculates the costs from the smallest subset of $\pi$ to the largest and subtracts the previous set's cost to obtain the marginal cost, i.e.,

$$\Delta_{\pi(j)} c(\pi) = c(P_{\pi(j)}(\pi) \cup \{\pi(j)\}) - c(P_{\pi(j)}(\pi)), \tag{13}$$

where $\pi(j)$ indicates the input in the $j$th location of $\pi$. For notational convenience, let $P_{\pi(j+1)}$ denote $P_{\pi(j)}(\pi) \cup \{\pi(j)\}$ for $0 < j < k$. We also assume the function $\eta(\cdot)$ puts the

**Algorithm 1.**

1. Choose $m, N_V, N_O$, and $N_I$; set $\widehat{Sh}_i = 0$ for $i = 1, 2, \ldots, k$ and $counter = 0$
2. For $q = 1, 2, \ldots, N_V$
   (a) Sample $\mathbf{X}_{\mathcal{K}}^{(q)}$ from $\mathbf{G}_{\mathcal{K}}$
   (b) Evaluate $Y^{(q)} = \eta(\mathbf{X}_{\mathcal{K}}^{(q)})$
3. Calculate $\bar{Y} = {N_V}^{-1} \sum_{q=1}^{N_V} Y^{(q)}$ and $\widehat{\mathrm{Var}}[Y] = (N_V - 1)^{-1} \sum_{q=1}^{N_V} (Y^{(q)} - \bar{Y})^2$
4. While $counter < m$
   (a) Generate $\pi \in \Pi(k)$
   (b) Set $prevC = 0$
   (c) For $j = 1, 2, \ldots, k$
      i. If $j = k$
         $\widehat{c}(P_{\pi(j)}(\pi) \cup \{\pi(j)\}) = \widehat{\mathrm{Var}}[Y]$
      ii. Else \\ comment: $0 < j < k$
         A. For $l = 1, 2, \ldots, N_O$
            Sample $\mathbf{X}^{(l)}_{-P_{\pi(j+1)}(\pi)}$ from $\mathbf{G}_{-P_{\pi(j+1)}(\pi)}$
            For $h = 1, 2, \ldots, N_I$
               Sample $\mathbf{X}^{(l,h)}_{P_{\pi(j+1)}(\pi)}$ from $\mathbf{G}_{P_{\pi(j+1)}(\pi)\left|\mathbf{X}^{(l)}_{-P_{\pi(j+1)}(\pi)}\right.}$
               Evaluate $Y^{(l,h)} = \eta\left(\mathbf{X}^{(l,h)}_{P_{\pi(j+1)}(\pi)}, \mathbf{X}^{(l)}_{-P_{\pi(j+1)}(\pi)}\right)$
            Calculate $\bar{Y}^{(l)} = {N_I}^{-1} \sum_{h=1}^{N_I} Y^{(l,h)}$
            Calculate $\widehat{\mathrm{Var}}\left[Y|\mathbf{X}^{(l)}_{-P_{\pi(j+1)}(\pi)}\right] = (N_I - 1)^{-1} \sum_{h=1}^{N_I} (Y^{(l,h)} - \bar{Y}^{(l)})^2$
         B. Calculate $\widehat{c}(P_{\pi(j+1)}(\pi)) = {N_O}^{-1} \sum_{l=1}^{N_O} \widehat{\mathrm{Var}}\left[Y|\mathbf{X}^{(l)}_{-P_{\pi(j+1)}(\pi)}\right]$
      iii. Calculate $\widehat{\Delta}_{\pi(j)} c(\pi) = \widehat{c}(P_{\pi(j+1)}(\pi)) - prevC$
      iv. Update $\widehat{Sh}_{\pi(j)} = \widehat{Sh}_{\pi(j)} + \widehat{\Delta}_{\pi(j)} c(\pi)$
      v. Set $prevC = \widehat{c}(P_{\pi(j+1)}(\pi))$
   (d) $counter = counter + 1$
5. $\widehat{Sh}_i = \widehat{Sh}_i / m$ for $i = 1, 2, \ldots, k$

inputs in the right order, i.e., $\eta(\mathbf{X}_{\{1,3\}}, X_2) = \eta(X_1, X_2, X_3)$. Note that our cost function $c(\mathcal{J})$ involves a (conditional) variance of a complicated function $\eta(\mathbf{X}_{\mathcal{K}})$, which Algorithm 1 estimates by Monte Carlo simulation. Three parameters of the Monte Carlo simulation are introduced in Algorithm 1: the number of Monte Carlo samples $N_V$ to estimate $\mathrm{Var}[Y]$, the number of outer samples $N_O$, and the number of inner samples $N_I$ at each outer sample to estimate $\mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-\mathcal{J}}]]$. Appendix B analyzes how these parameters affect the variance of the estimated Shapley effects and makes recommendations about how to choose these parameters.

Ignoring the estimation of $c$ for the moment, Algorithm 1 saves the cost of the current set to $prevC$ to calculate the next marginal cost in Step 4(c)v. For $\pi = \{1, 3, 2\}$, Algorithm 1 performs

$$\begin{aligned}
\Delta_{\pi(1)} c(\pi) &= c(\{1\}) - c(\emptyset) \quad \text{Set } prevC = 0, \\
\Delta_{\pi(2)} c(\pi) &= c(\{1, 3\}) - prevC, \quad \text{Set } prevC = c(\{1, 3\}), \\
\Delta_{\pi(3)} c(\pi) &= c(\{1, 3, 2\}) - prevC, \quad \text{Set } prevC = c(\{1, 3, 2\}),
\end{aligned}$$

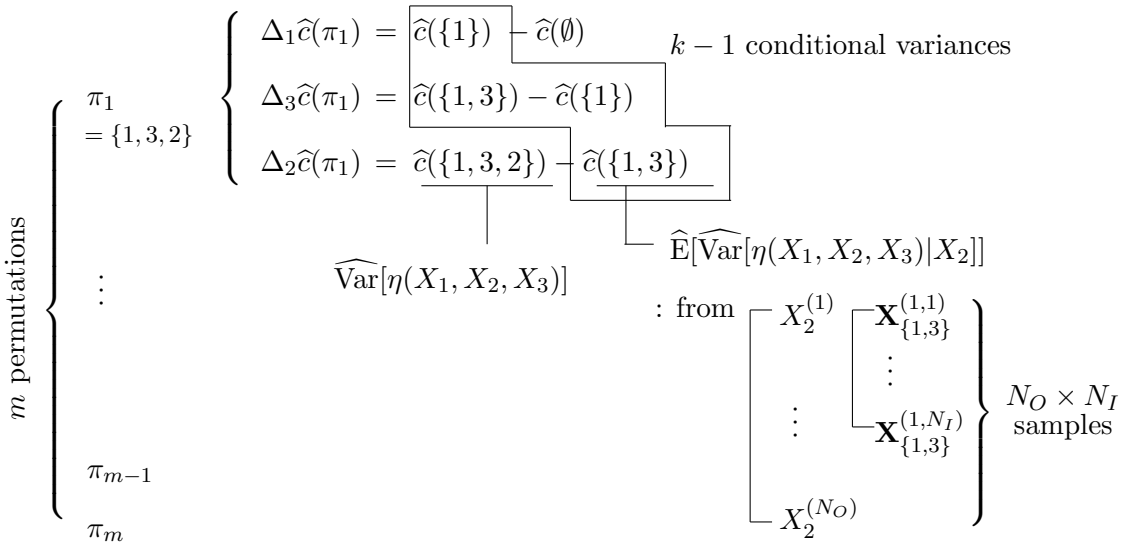



**Figure 1.** *Illustration of allocation of the computation budget.*

which only requires three cost function evaluations. Therefore, Algorithm 1 reduces the computational load of ApproShapley by half.

Taking the estimation of $c$ into account, the total computation budget to estimate $Sh_i$ grows by the factor of the sampling budget for the Monte Carlo variance estimation, which can be a substantial increase. Figure 1 illustrates estimation of the Shapley effects when $k = 3$. Notice that the costs are variances estimated from Monte Carlo simulations. For each sampled permutation $\pi_\ell$ we need $k$ estimated costs; one total variance $\mathrm{Var}[Y]$ and $\mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-P_{\pi_\ell(j)}}]]$ for $j = 2,\ldots,k$. Shown in Figure 1 is the case when $\pi_1 = \{1,3,2\}$; we need one $\widehat{\mathrm{Var}}[Y]$ for $c(\{1,3,2\})$ and conditional variances for $c(\{1\})$ and $c(\{1,3\})$, respectively. Figure 1 shows estimation of $c(\{1,3\}) = \mathrm{E}[\mathrm{Var}[\eta(X_1,X_2,X_3)|X_2]]$ as an example. The outer samples $X_2^{(1)}, X_2^{(2)}, \ldots, X_2^{(N_O)}$ are generated first, and then conditional on each $X_2^{(l)}$, $\mathbf{X}_{\{1,3\}}^{(l,1)}, \mathbf{X}_{\{1,3\}}^{(l,2)}, \ldots, \mathbf{X}_{\{1,3\}}^{(l,N_I)}$ are sampled. See section 4.2 for further discussion on conditional sampling. Each inner-outer sample pair is used to evaluate $\eta(\cdot)$. Thus, the total computational budget for Algorithm 1 is $N_V + mN_IN_O(k-1)$.

Before performing a sensitivity analysis, we would typically estimate $\mathrm{Var}[Y]$ as precisely as possible to measure the overall variability in the output due to the uncertain inputs. If $\widehat{\mathrm{Var}}[Y]$ is negligible compared to the required tolerance for the system, then it can be considered insensitive and further sensitivity analysis would no longer be of interest. If $\widehat{\mathrm{Var}}[Y]$ is significant, then we would proceed to estimate the Shapley effects and $\widehat{\mathrm{Var}}[Y]$ can be used in the estimation. To provide a good estimator of $\mathrm{Var}[Y]$, $N_V$ should be chosen sufficiently large.

Given the computation budget, there is a trade-off between sampling more permutations (large $m$) and more precise estimation of the cost functions (large $N_I$ and $N_O$). A reasonable

criterion for choosing the values of these parameters is to minimize $\mathrm{Var}[\widehat{Sh}_i]$. In Appendix B, we show that it is an effective strategy to choose $N_I = 3, N_O = 1$, and $m$ as large as possible given these choices.

When the computation budget can be chosen to ensure a desired estimation quality of the Shapley effects, then we recommend performing sequential experiments by increasing $m$ while maintaining $N_I = 3$ and $N_O = 1$. Notice that $\widehat{Sh}_i$ is a sample mean of $\widehat{\Delta}_i c(\pi)$'s and therefore the standard error of $\widehat{Sh}_i$ can be easily estimated from the sample variance of $\widehat{\Delta}_i c(\pi)$. Since $\mathrm{Var}[\widehat{Sh}_i] = O(m^{-1})$ (see Appendix B), the standard errors of the Shapley effects are $O(m^{-1/2})$. For example, if the estimated relative error of $\widehat{Sh}_1$ is 10% when $m = 5{,}000$, then we can decrease it to approximately 5% by increasing $m$ to 20,000.

This example also implies that the number of permutations $m$ needed to obtain the desired standard errors of the Shapley effects is not directly related to the number of inputs $k$; it is rather related to the variance of the model response $Y = \eta(\mathbf{X}_{\mathcal{K}})$. The following theorem shows that the variance of $\widehat{Sh}_i$ has an upper bound that is independent of $k$. The proof can be found in Appendix A.

Theorem 3. *If $c$ is evaluated exactly, then* $\mathrm{Var}[\widehat{Sh}_i] \leq (\mathrm{Var}[Y])^2/m$.

Theorem 3 shows that if we add a number of dummy inputs that do not affect the output $Y$, the variance of the estimated Shapley effects would have the same bound as before. Hence, even if $k$ is large, if $\mathrm{Var}[Y]$ is small, $m$ does not need to be too large to reduce $\mathrm{Var}[\widehat{Sh}_i]$ to an acceptable level.

Of course, if $k$ is small, e.g., $k = 3$, then the number of possible permutations is small ($3! = 6$), in which case we would rather examine all possible permutations as opposed to randomly sampling them and spend more of the budget to estimate the cost functions. Therefore, step 4(a) in Algorithm 1 can be modified to go through each permutation exactly once, which gets rid of the uncertainty in $\widehat{Sh}_i$ from randomly sampling permutations. In this case, we recommend $N_I = 3$ and setting $N_O$ to consume the remaining budget (see Appendix B).

The first-order and total effects can also be estimated by Algorithm 1 using the semivalue expressions in (8) and (9), respectively. The estimator of $V_i$ is the sample average of $\widehat{\Delta}_i c(\pi)$'s when $i$ is the last element of the permutation $\pi$. The estimator of $T_i$ is the sample average of $\widehat{\Delta}_i c(\pi)$'s when $i$ is the first element of the permutation $\pi$. Therefore, the numbers of samples that are used to estimate $V_1, V_2, \ldots, V_k$ (or $T_1, T_2, \ldots, T_k$) are random. However, each input has probability $1/k$ of being the last (or the first) element in each permutation. Therefore, the expected number of samples available to estimate each $V_i$ or $T_i$ is $m/k$ for all $X_i$'s.

Saltelli [20] proposes a more efficient estimation method of the first-order and total effects than two-level Monte Carlo simulation for the case of independent inputs. His method stores the sampled input values in two $N \times k$ matrices where each of $k$ columns corresponds to each input, then creates a matrix by replacing the $i$th column of the second matrix with the $i$th column of the first matrix. Therefore, the resulting matrix and the first matrix share the same $i$th column, which can be seen as two independent sets of $N$ samples of inputs conditional on $X_i$. Using this shortcut, the method can estimate the first-order and total effects using

only $N(k+2)$ evaluations of $\eta(\cdot)$. Clearly, this method is no longer valid when the inputs are dependent.

**4.2. Generating dependent inputs.** Step 4(c)ii of Algorithm 1 involves conditional sampling of inputs. When there is dependence among inputs, then we need to correctly define the conditional distribution from which to sample.

A popular method when there is correlation among inputs is copula modeling, which defines the relationship of $\mathbf{G}_{\mathcal{K}}$ and the marginals, $G_1, G_2, \ldots, G_k$, as

$$\mathbf{G}_{\mathcal{K}}(x_1, x_2, \ldots, x_k) = C(G_1(x_1), G_2(x_2), \ldots, G_k(x_k)),$$

where $C$ is a copula (Nelsen [16]). As long as the correlation structure among inputs can be modeled by a copula, we can easily sample the correlated inputs in Algorithm 1.

Cario and Nelson [4] develop ARTA (AutoRegressive To Anything), which obtains the desired autoregressive process by transforming a standardized Gaussian autoregressive process with a carefully chosen autocorrelation structure. In examples in section 5, we use NORTA (NORmal To Anything) suggested by Cario and Nelson [5] to generate correlated inputs. NORTA, a generalization of ARTA, finds the correlation matrix of a $k$-dimensional standard multivariate normal vector $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_k\}$ such that

$$\mathbf{X}_{\mathcal{K}} = \left[G_1^{-1}(\Phi(Z_1)), G_2^{-1}(\Phi(Z_2)), \ldots, G_k^{-1}(\Phi(Z_k))\right]^{\top}, \tag{14}$$

where $\Phi$ is the standard normal cumulative distribution function. The resulting $\mathbf{X}_{\mathcal{K}}$ in (14) has the desired correlation structure as well as the marginal distribution for each $X_i$. For the restrictions on $\mathbf{G}_{\mathcal{K}}$ to satisfy (14), see Cario and Nelson [5]. In our examples, we impose a rank correlation between two continuous dependent inputs, then generate a standard bivariate normal vector that has the same rank correlation, which we transform to the inputs using (14). Since rank correlation is invariant under strictly increasing transformations, the resulting inputs have the intended rank correlation.

**5. Numerical experiments.** In this section, we present two examples where we estimate the Shapley effects and compare them with the first-order and total effects. The first example is a simple make-to-order manufacturing system where we have multiple product types modeled as a Jackson network (Jackson [14]). Queueing network models are widely used in industrial engineering and operations research to design and improve manufacturing and service systems. In our example, the order arrival rates, correlated according to the substitutability and complementarity of the products, change according to random distributions every month. The goal of the sensitivity analysis is to find the type of product that causes the largest fluctuation in the expected order completion time of all jobs. This information may assist the company to manage tension in the manufacturing system to reduce the fluctuation.

The second example is sensitivity of fire-spread in the Mediterranean shrublands using the model in Salvador et al. [25]. The model has 10 environmental variables representing the physical state of the shrubland. Salvador et al. [25] assume these inputs are independent; however, the data analysis of Clark et al. [10] shows that some of the inputs are correlated. We take this correlation structure into account in our sensitivity analysis and identify the impacts of the factors on the speed of fire-spread.

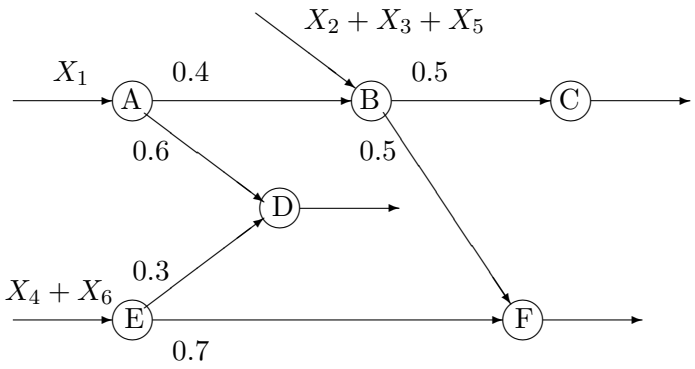



**Figure 2.** *A queueing network with six independent arrival processes.*

**5.1. Manufacturing system model.** Suppose the network in Figure 2 represents a manufacturing line that has six workstations, A–F, and processes six different types of jobs. A job arrives to the manufacturing line once a customer places an order for the corresponding product type. The daily order arrival rates of six jobs, $X_1, X_2, \ldots, X_6$, remain constant throughout a month, fluctuating each month. The monthly expected job completion time of this manufacturing line is the output of the model, and the input $\mathbf{X}$ is the vector of six arrival rates. In terms of supply chain management, the fluctuation in this expected time may incur cost to the company. Under this premise, we apply sensitivity analysis to identify the types of jobs whose arrival rates affect the variation in the expected completion time the most.

The first workstation where each job is processed depends on the job type; type 1 starts at workstation A, types 2, 3, and 5 start at workstation B, and types 4 and 6 start at workstation E. After being processed at its initial station, the job is routed to the next station. Figure 2 shows the routing probabilities from one station to the others. Each station has one machine that processes multiple types of jobs. The interarrival times and processing times are exponentially distributed and independent from each other. The job processing rates at stations, denoted by $\mu_A, \mu_B, \ldots, \mu_F$, are $1.2, 1.5, 4, 1.8, 3.6, 1.5$ jobs per day, respectively. We assume that the service rates can be set by the manufacturer and therefore are decision variables and are not included in the global sensitivity analysis.

For all six arrival rates, we used the same marginal distribution whose minimum, maximum, and mode are 0.5, 0.8, and 0.6, respectively. This type of distribution is typically modeled with a beta distribution in practice, which we also adopt here. Suppose products of types 1 and 2 are complementary, and therefore their demands are positively correlated. Products of types 3 and 4, on the other hand, substitute for each other, which makes their demands negatively correlated. The rank correlations among the arrival rates are $\mathrm{Corr}(X_1, X_2) = 0.5$ and $\mathrm{Corr}(X_3, X_4) = -0.5$. The correlated inputs are generated using NORTA as described in section 4.2.

The Jackson network model of Figure 2, which can be found in Appendix C, provides the expected completion time of a job $Y$ as a function of $\mathbf{X}$. In our problem, these arrival rates are the random inputs, which cause the expected job completion to be random. Table 1

**Table 1**

*Sensitivity measures of the expected job completion time in hours with standard errors in parentheses.*

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | Sum |
|---|---|---|---|---|---|---|---|
| $Sh_i$ | 4.23 | 2.79 | 0.62 | 2.45 | 1.37 | 3.88 | 15.34 |
| | (0.07) | (0.07) | (0.03) | (0.03) | (0.02) | (0.02) | |
| $V_i$ | 5.91 | 4.43 | 0.15 | 1.85 | 1.40 | 3.76 | 17.50 |
| | (0.05) | (0.05) | (0.07) | (0.06) | (0.06) | (0.05) | |
| $T_i$ | 2.54 | 1.05 | 1.07 | 2.94 | 1.42 | 4.04 | 13.06 |
| | (0.01) | (0.01) | (0.01) | (0.01) | (0.01) | (0.02) | |

shows the estimated first-order, total, and Shapley effects of all inputs. All three effects are calculated from Algorithm 1 in one experiment using $N_V = 2{,}000, N_O = 100$, and $N_I = 2$ with all $m = 6!$ permutations of inputs.

The right-most column of Table 1 confirms Theorem 2 in section 3. The sum of the Shapley effects is the estimated variance of the output, which is the expected job completion time. In this example, the sum of the first-order effects is greater than the output variance, whereas the sum of the total effects is smaller than the output variance. Neither is desirable if we want to measure the sensitivity of output variance to arrival rates in absolute terms.

Suppose that the company wishes to know which two arrival rates are most influential on uncertainty in expected completion time. A common practice in sensitivity analysis is to use both first-order effects and total effects to measure sensitivity. The first-order effects and total effects disagree about which two arrival rates are most influential: $V_1$ and $V_2$ are the largest first-order effects, whereas $T_4$ and $T_6$ are the largest total effects. It would not be easy to combine the first-order effects and total effects to answer this question. One might rank all of the first-order effects and total effects. The largest two of the first-order effects and total effects are $V_1$ and $V_2$; the total effects lose out in this comparison. However, it is questionable whether there is a fair comparison between the actual values of a first-order effect and a total effect, considering that the sum of the first-order effects exceeds the total variance and the sum of the total effects is less than the total variance. The Shapley effects identify the arrival rates $X_1$ and $X_6$ as most influential on the output variance. In this way the Shapley effects succeed in resolving which two of the four rates identified by the first-order and total effects are really the largest contributors to $\mathrm{Var}[Y]$—and therefore to supply chain cost—allowing the manufacturer to take the the most effective actions to mitigate their impact.

**5.2. Fire-spread model.** The model in Salvador et al. [25] is based on Rothermel's fire-spread model (Rothermel [19]); its primary output is the rate of the spread of the front point of a fire. Salvador et al. [25] also adopted the modifications by Albini [1] on the net fuel loading and the optimum reaction velocity, and the modifications by Catchpole and Catchpole [8] on the moisture damping coefficient and the heat of preignition. The complete set of equations used in this section is provided in Appendix D.

The modified model has the 10 inputs given in Table 2, which also provides the distributions used in our sensitivity analysis experiment. Salvador et al. [25] only considered the fuels with a diameter less than 6 mm as they have the highest influence on the fire behavior. They assumed all inputs in Table 2 are independent. However, the data analysis of Clark

**Table 2**

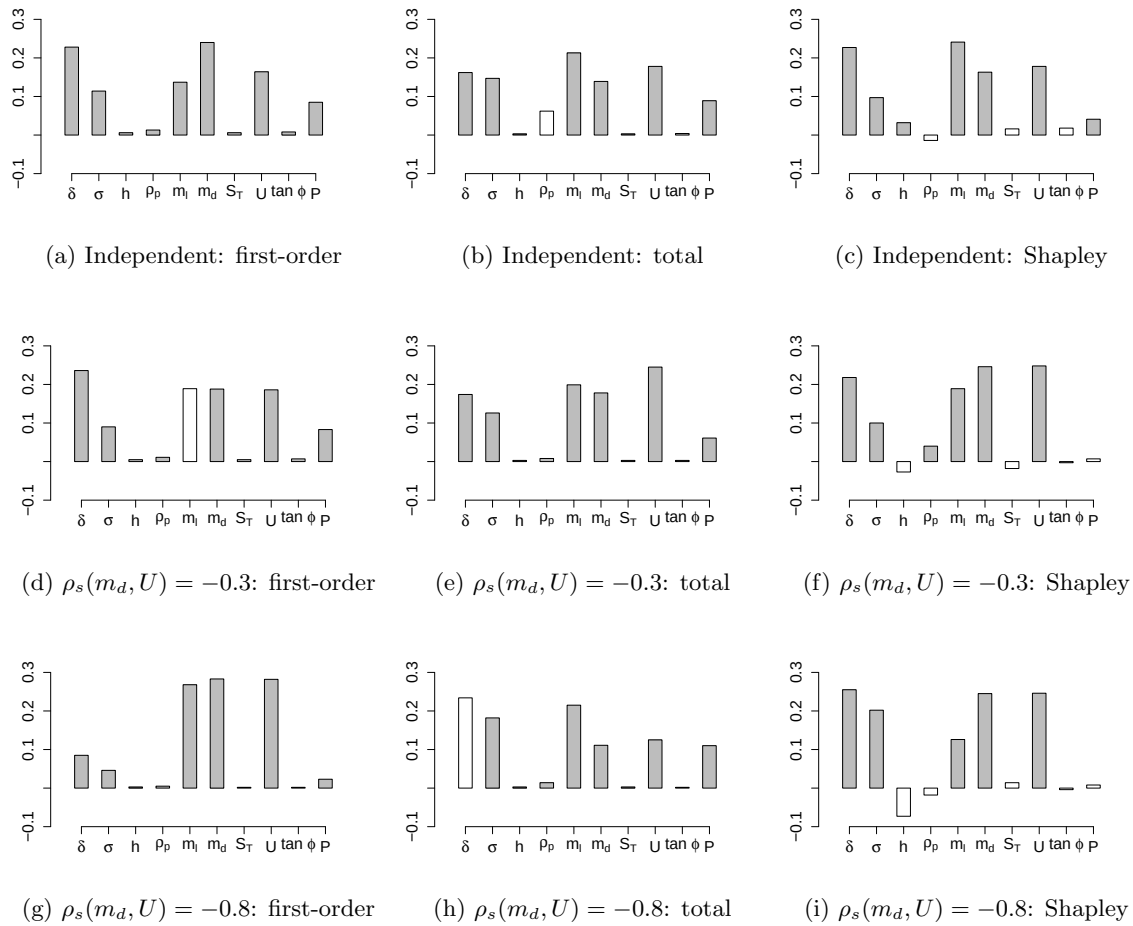
*Distributions of* 10 *inputs of the fire-spread model. D.W.: dry weight; MIN.: mineral weight;* $\mathrm{N}(\mu, \sigma)$*: the normal distribution with mean* $\mu$ *and standard deviation* $\sigma$*;* $a\,\mathrm{LogN}(\mu, \sigma)$*: the distribution of* $a\exp(Z)$ *where* $Z$ *has a normal distribution with mean* $\mu$ *and standard deviation* $\sigma$*.*

| | Input | Symbol and unit | Distribution |
|---|---|---|---|
| 1 | Fuel depth | $\delta$ (cm) | $\mathrm{LogN}(2.19, 0.517)$ |
| 2 | Fuel particle area-to-volume ratio | $\sigma$ ($\mathrm{cm}^{-1}$) | $\mathrm{LogN}(3.31, 0.294)$ |
| 3 | Fuel particle low heat content | $h$ ($\mathrm{Kcal\,kg}^{-1}$) | $\mathrm{LogN}(8.48, 0.063)$ |
| 4 | Oven-dry particle density | $\rho_p$ ($\mathrm{D.W.g\,cm}^{-3}$) | $\mathrm{LogN}(-0.592, 0.219)$ |
| 5 | Moisture content of the live fuel | $m_l$ ($\mathrm{H_2O\,g\,D.W.g}^{-1}$) | $\mathrm{N}(1.18, 0.377)$ |
| 6 | Moisture content of the dead fuel | $m_d$ ($\mathrm{H_2O\,g\,D.W.g}^{-1}$) | $\mathrm{N}(0.19, 0.047)$ |
| 7 | Fuel particle total mineral content | $S_T$ ($\mathrm{MIN.\,g\,D.W.g}^{-1}$) | $\mathrm{N}(0.049, 0.011)$ |
| 8 | Wind speed at midflame height | $U$ ($\mathrm{km\,h}^{-1}$) | $6.9\,\mathrm{LogN}(1.0174, 0.5569)$ |
| 9 | Slope | $\tan\phi$ | $\mathrm{N}(0.38, 0.186)$ |
| 10 | Dead fuel loading to total fuel loading | $P$ | $\mathrm{LogN}(-2.19, 0.64)$ |

et al. [10] shows that there is a negative rank correlation ($-0.355$) between the wind speed and the moisture content of the dead fuel, i.e., the windier it is, the less moisture the dead fuels contain. For the purpose of our experiment, three cases were tested: (1) independent inputs, (2) a mild correlation (rank correlation: $-0.3$) between $m_d$ and $U$, and (3) a stronger correlation (rank correlation: $-0.8$) between $m_d$ and $U$. All inputs except for the fuel depth ($\delta$), the moisture content of the dead fuel ($m_d$), and the wind speed at midflame height ($U$) have the same marginal distributions as in Salvador et al. [25]. We moved the mean of the distribution of the fuel depth ($\delta$) from 24.3 to 10.2 to see the impact of the correlated inputs on the sensitivity measures more clearly. Empirical distributions of $m_d$ and $U$ are not provided in Salvador et al. [25], but the means and the coefficients of variation are. Therefore, we used normal and lognormal distributions for $m_d$ and $U$, respectively, matching their means and the coefficients of variation. Notice that we tested their "strong wind" scenario in which the wind speed $U$ is multiplied by 6.9.

The correlated inputs are generated using NORTA as described in section 4.2. All negative values of $m_\ell, S_T$, and $\tan\phi$ are rejected as in Salvador et al. [25]. For $P$, a ratio of dead fuel loading to total fuel loading, values over 1 are rejected. Also, any input values of $\sigma$ less than $3/0.6$ are rejected as $3/0.6$ is the smallest possible surface area to volume ratio for fuels with a diameter less than 6 mm.

Figure 3 shows the estimated measures of the sensitivity of the output (the rate of the spread of the front point of a fire) to the inputs. Notice that each effect is normalized by the sum of effects of all inputs. The Shapley effects are estimated from $m = 1{,}500{,}000$ permutations with $N_V = 100{,}000, N_I = 3$, and $N_O = 1$. The first-order and total effects are estimated directly from their definitions, $V_i = \mathrm{Var}[\mathrm{E}[\eta(\mathbf{X}_\mathcal{K})|X_i]]$ and $T_i = \mathrm{E}[\mathrm{Var}[\eta(\mathbf{X}_\mathcal{K})|\mathbf{X}_{-i}]]$, by two-level Monte Carlo simulation. Each effect is estimated from 680 inner samples and 680 outer samples, which gives approximately the same budget for the Shapley effects estimation;




(a) Independent: first-order (b) Independent: total (c) Independent: Shapley

(d) $\rho_s(m_d, U) = -0.3$: first-order (e) $\rho_s(m_d, U) = -0.3$: total (f) $\rho_s(m_d, U) = -0.3$: Shapley

(g) $\rho_s(m_d, U) = -0.8$: first-order (h) $\rho_s(m_d, U) = -0.8$: total (i) $\rho_s(m_d, U) = -0.8$: Shapley

**Figure 3.** *Sensitivity measures of inputs of the fire-spread model normalized by the sum of effects of all inputs. White bars indicate the estimated effects are within two standard errors from* 0.

$680 \times 680 \times 10 \approx 1{,}500{,}000 \times 3 + 100{,}000$. Standard errors of the effects are also estimated; a batch of size 68 (10 batches) is used to provide an estimate of $\mathrm{Var}[V_i]$ for each $i$ (see Asmussen and Glynn [2]).

The colors of bars in Figure 3 indicate statistical significance; a white bar indicates that the estimated effect is within two standard errors from 0. Notice that the estimated Shapley effects that are statistically insignificant are the ones to which the response $Y$ is less sensitive, e.g., the estimated Shapley effects of $S_T$, and $\tan\phi$ in the independent input case.

Some of the estimated Shapley effects are negative. This is because the estimator $\widehat{Sh}_i$ from Algorithm 1 is a sum of differences of the cost functions and therefore, if $Sh_i$ is close to 0, $\widehat{Sh}_i$ can be negative due to the estimation error. Notice that the Shapley effects with negative estimates in Figure 3 are statistically insignificant. The possibility for an estimate of a variance component to be negative is familiar from hierarchical or mixed models in statistics.

In all three settings of independent/dependent inputs, the Shapley effects help us decide which inputs are more influential when the first-order and total effects identify different sets

of inputs to be influential. For instance, in the independent input case, we would conclude that $\delta$ and $m_d$ have more impact on $\mathrm{Var}[Y]$ than other inputs based on the first-order effects (Figure 3a) or that $m_l$ and $U$ are more important based on the total effects (Figure 3b). The Shapley effects (Figure 3c) indicate $\delta$ and $m_l$ are more important, which is the middle ground between the first-order and total effects. As $m_d$ and $U$ become highly correlated ($\rho_s(m_d, U) = -0.8$), there is more discrepancy between the first-order and total effects. The first-order effects (Figure 3g) show that $U$, $m_d$, and $m_l$ have bigger impacts on $\mathrm{Var}[Y]$ than others, whereas the total effects (Figure 3h) indicate that $\delta$, $m_l$, and $\sigma$ are more important. From the Shapley effects (Figure 3i), we can conclude that $\delta, m_d, U$, and $\sigma$ have more significant impact on $\mathrm{Var}[Y]$.

**6. Conclusions and future research.** In this paper we identified different purposes for global sensitivity analysis for which we would like to decompose the variance of the output into each input's contribution. We adopted Owen's sensitivity measure based on the concept of the Shapley value in game theory to define the Shapley effect. We established the connection between the first-order, total, and Shapley effects using the concept of semivalues. Exploiting the fact that the Shapley value is the only semivalue that sums to the total cost of all players in the game regardless of the cost function, we show the Shapley effects always decompose the total variance and allocate to each input regardless of the dependence among inputs or the structure of the output function, whereas the other two measures do not.

We examined the performance of all three measures using a simple manufacturing model and a realistic fire-spread model, both involving correlated inputs. In both experiments, the first-order and total effects identify different subsets of inputs as highly influential, and it would be difficult for a practitioner to use both measures to identify a small subset of highly influential inputs. The disagreement between first-order and total effects occurs because they consider different sets of interactions between inputs; neither considers all interactions. The Shapley effects, which take proper account of all interactions between inputs both structural and statistical, were useful to identify a small subset of highly influential inputs.

While the estimation of the Shapley effects may seem computationally prohibitive when the number of inputs is large, our proposed algorithm makes the Shapley effects estimable for any number of inputs. The algorithm consists of sampling the permutations of the inputs randomly and estimating the cost functions via Monte Carlo simulation. We also suggested a reasonable allocation of the computation budget to reduce the variance of the estimated Shapley effect. We proved that if the cost function can be calculated exactly, then the variance of the estimator of the Shapley effect is bounded above by a function of the number of sampled permutations, not the number of the inputs. Since we actually have to estimate the cost function in general, the variance may be larger than $(\mathrm{Var}[Y])^2/m$; however, it is still in $O(m^{-1})$ and independent of the number of the inputs.

As future research, we suggest Gaussian process emulation to deal with the case in which the computer model is costly to run. If $\eta(\cdot)$ is costly to evaluate, then an emulator $\widehat{\eta}(\cdot)$ can be used as a stand-in for $\eta(\cdot)$ to reduce the computation cost. If an emulator is used, the effort to estimate $c(\cdot)$ is effectively just the cost of the random-variate generation of $\mathbf{X}_{\mathcal{K}}$. The quality of this approximation depends on the accuracy of the model $\widehat{\eta}(\cdot)$ as it is based on a limited number of observations of the true surface $\eta(\cdot)$. Many observations may be required to get

an accurate Gaussian process emulator when the number of inputs is large. If the number of inputs is large, it could be valuable to conduct a preliminary study of the sensitivity to inputs. Then some inputs could be screened out before computing Shapley effects of the remaining inputs, or the preliminary study could enable a more efficient design of the experiment to build the Gaussian process emulator.

We may also want to fully account for the uncertainty in $\widehat{\eta}(\cdot)$ instead of simply plugging it in for $\eta(\cdot)$. Oakley and O'Hagan [17] provide an expression for first-order and total effects when a Gaussian process emulator is used as a stand-in for $\eta(\cdot)$. Taking a Bayesian point of view, they represent the mean vector and the covariance matrix of $\widehat{\eta}(\cdot)$ as functions of hyperparameters with prior distributions. After observing the functional values of $\eta(\cdot)$ at a number of design points, $\widehat{\eta}(\mathbf{X}_{\mathcal{K}})$ can be represented as a Gaussian process whose parameters have posterior distributions. In section 3.3 of Oakley and O'Hagan [17], the authors provide a formula for the posterior mean of the cost function $\tilde{c}$, i.e., $\mathrm{E}^{\star}[\mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{\mathcal{J}}]]]$ for any $\mathcal{J} \subset \mathcal{K}$, where $\mathrm{E}^{\star}$ is the expectation with respect to the posterior distribution of $\widehat{\eta}(\mathbf{X}_{\mathcal{K}})$. As shown in Theorem 1, $\tilde{c}$ or $c$ can be used to define the Shapley effect. Since $Sh_i$ is a weighted sum of $\mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{\mathcal{J}}]]$ for all $\mathcal{J} \subset \mathcal{K}$, we can obtain the posterior mean of $Sh_i$ from Oakley and O'Hagan's expression [17]. Hence, we may replace the cost estimation in step 4(c)ii with calculating the posterior mean of the cost.

The expression for the posterior mean of the cost function includes integrations of products of means and variances of the Gaussian process. Typically, these integrations do not have closed-form expressions for a general choice of $\mathbf{G}_{\mathcal{K}}$. We can numerically integrate them by Monte Carlo simulation, which costs some computational effort. However, if the original $\eta(\cdot)$ is very expensive to evaluate, these Monte Carlo integrations will still be substantially less costly.

**Appendix A. Proofs of theorems.** For completeness, we restate the theorems with their proofs.

Theorem 1. *The Shapley values defined using cost function $\tilde{c}$ and $c$ are equivalent.*

*Proof.* For $\mathcal{J} \subseteq \mathcal{K}\backslash\{i\}$

$$\begin{aligned}
\tilde{c}(\mathcal{J}\cup\{i\}) - \tilde{c}(\mathcal{J}) &= \mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{\mathcal{J}\cup\{i\}}]] - \mathrm{Var}[\mathrm{E}[Y|\mathbf{X}_{\mathcal{J}}]] \\
&= (\mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{\mathcal{J}\cup\{i\}}]]) - (\mathrm{Var}[Y] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{\mathcal{J}}]]) \\
&= \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{\mathcal{J}}]] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{\mathcal{J}\cup\{i\}}]] \\
&= \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-(\mathcal{A}\cup\{i\})}]] - \mathrm{E}[\mathrm{Var}[Y|\mathbf{X}_{-\mathcal{A}}]] \\
&= c(\mathcal{A}\cup\{i\}) - c(\mathcal{A}), \qquad (15)
\end{aligned}$$

where $\mathcal{A} = \mathcal{K}\backslash(\mathcal{J}\cup\{i\})$. The second equality follows from the law of total variance. Notice that for all $\mathcal{J} \subseteq \mathcal{K}\backslash\{i\}$, the corresponding set $\mathcal{A}$ is unique. When calculating the Shapley effect from (4) using $\tilde{c}$, $\tilde{c}(\mathcal{J}\cup\{i\}) - \tilde{c}(\mathcal{J})$ is weighted by $\frac{(k-|\mathcal{J}|-1)!|\mathcal{J}|!}{k!}$. Since $|\mathcal{A}| + |\mathcal{J}| + 1 = k$,

$$\frac{(k-|\mathcal{J}|-1)!|\mathcal{J}|!}{k!} = \frac{|\mathcal{A}|!(k-|\mathcal{A}|-1)!}{k!},$$

which is the weight for $c(\mathcal{A}\cup\{i\}) - c(\mathcal{A})$ to calculate the Shapley value using $c$. Therefore, $c$ and $\tilde{c}$ give exactly the same Shapley values. ∎

Theorem 2. *There exists a joint distribution* $\mathbf{G}_{\mathcal{K}}$ *and function* $\eta(\cdot)$ *such that* $\sum_{i=1}^{k} V_i > \mathrm{Var}[Y] > \sum_{i=1}^{k} T_i$.

*Proof.* Consider the following simple response function that has $k = 2$ inputs:

$$\eta(\mathbf{X}_{\mathcal{K}}) = X_1 + X_2, \tag{16}$$

where $\mathbf{X}_{\mathcal{K}} = \{X_1, X_2\}$ are jointly normally distributed with $\mathrm{E}[X_1] = \mathrm{E}[X_2] = 0$, $\mathrm{Var}[X_1] = \mathrm{Var}[X_2] = 1$, and $\mathrm{Corr}(X_1, X_2) = \rho$. From the definitions of the first-order and total effects in (1) and (2), we can obtain $V_1 = V_2 = (1+\rho)^2$ and $T_1 = T_2 = (1-\rho)^2$. Hence, it is easy to see that for $\rho > 0$, $V_1 + V_2 = 2(1+\rho)^2 > \mathrm{Var}[Y] = 2(1+\rho) > 2(1-\rho^2) = T_1 + T_2$. ∎

Theorem 3. *If* $c$ *is evaluated exactly, then* $\mathrm{Var}[\widehat{Sh}_i] \le (\mathrm{Var}[Y])^2/m$.

*Proof.* Since we assumed $c$ is evaluated exactly, the variance in $\widehat{Sh}_i$ comes from the random sampling of permutations in Algorithm 1. Notice that $\widehat{Sh}_i$ is a sample mean of cost differences $\Delta_i c(\pi_\ell) = c(P_i(\pi_\ell) \cup \{i\}) - c(P_i(\pi_\ell))$ for $\ell = 1, 2, \ldots, m$. As $0 \le c(\mathcal{J}) \le \mathrm{Var}[Y]$ for all $\mathcal{J} \subseteq \mathcal{K}$, $\Delta_i c(\pi_\ell)$ is strictly bounded between $-\mathrm{Var}[Y]$ and $\mathrm{Var}[Y]$. The variance of $\Delta_i c(\pi_\ell)$ is maximized when it takes the values $-\mathrm{Var}[Y]$ and $\mathrm{Var}[Y]$ with probability $1/2$, each. Hence, $\mathrm{Var}[\Delta_i c(\pi_\ell)] \le (-\mathrm{Var}[Y])^2/2 + (\mathrm{Var}[Y])^2/2 = (\mathrm{Var}[Y])^2$, which leads to the conclusion. ∎

**Appendix B. Optimal budget allocation for Algorithm 1.** Assume $N_V$ simulations have already been conducted to estimate $\mathrm{Var}[Y]$. Then we have computation budget $B$ that we can allocate to $m$, $N_I$, and $N_O$ so that $B = mN_IN_O(k-1)$ subject to $N_I \ge 2$ to obtain the sample variance. Our criterion to allocate the budget is to minimize the variance of the estimated Shapley effects. We consider two cases: (a) when $k$ is small, and therefore, all $k!$ permutations are considered, and (b) when $k$ is large and we randomly sample permutations.

As metioned in section 4.1, if $k$ is small, then we should examine all possible permutations instead of randomly sampling them. In this case the budget is divided among $N_I$ and $N_O$, i.e., $B = k!N_IN_O(k-1)$.

Claim 1. *When all* $k!$ *permutations are examined in Algorithm* 1*, the near-optimal allocation of the computation budget is* $N_I = 3$ *and* $N_O = C/3$ *for some constant* $C$.

*Proof.* Without loss of generality, assume we aim to minimize $\mathrm{Var}[\widehat{Sh}_1]$. The estimator $\widehat{Sh}_1$ from Algorithm 1 when we exhaust all $k!$ permutations is

$$\widehat{Sh}_1 = \frac{1}{k!} \sum_{\pi \in \Pi(\mathcal{K})} (\widehat{c}(P_1(\pi) \cup \{1\}) - \widehat{c}(P_1(\pi))).$$

Since all costs are estimated independently,

$$\mathrm{Var}[\widehat{Sh}_1] = \frac{1}{(k!)^2} \sum_{\pi \in \Pi(\mathcal{K})} (\mathrm{Var}[\widehat{c}(P_1(\pi) \cup \{1\})] + \mathrm{Var}[\widehat{c}(P_1(\pi))]). \tag{17}$$

Recall that

$$\widehat{c}(P_1(\pi)) = \widehat{\mathrm{E}}[\widehat{\mathrm{Var}}[Y|\mathbf{X}_{-P_1(\pi)}]] = \frac{1}{N_O} \sum_{l=1}^{N_O} \frac{1}{N_I - 1} \sum_{h=1}^{N_I} (Y^{(l,h)} - \bar{Y}^{(l)})^2,$$

where $Y^{(l,h)} = \eta(\mathbf{X}^{(l,h)}_{P_1(\pi)}, \mathbf{X}^{(l)}_{-P_1(\pi)})$ and $\bar{Y}^{(l)} = {N_I}^{-1}\sum_{h=1}^{N_I} Y^{(l,h)}$. Letting $\widehat{V}_l = (N_I-1)^{-1}\sum_{h=1}^{N_I}(Y^{(l,h)} - \bar{Y}^{(l)})^2$, $\widehat{c}(P_1(\pi_\ell))$ is simply a sample average of $\widehat{V}_1, \widehat{V}_2, \ldots, \widehat{V}_{N_O}$. Therefore,

$$\begin{aligned}
\mathrm{Var}[\widehat{c}(P_1(\pi))] &= \mathrm{Var}\left[\frac{1}{N_O}\sum_{l=1}^{N_O}\widehat{V}_l\right] \\
&= \frac{1}{N_O}\mathrm{Var}[\widehat{V}_l] \\
&= \frac{1}{N_O}\left(\mathrm{E}[\mathrm{Var}[\widehat{V}_l|\mathbf{X}^{(l)}_{-P_1(\pi)}]] + \mathrm{Var}[\mathrm{E}[\widehat{V}_l|\mathbf{X}^{(l)}_{-P_1(\pi)}]]\right) \\
&= \frac{1}{N_O N_I}\left(\mathrm{E}[M_4^{(l)}] - \frac{N_I-3}{N_I-1}\mathrm{E}[(M_2^{(l)})^2] + N_I\mathrm{Var}[M_2^{(l)}]\right), \qquad (18)
\end{aligned}$$

where $M_2^{(l)}$ and $M_4^{(l)}$ are the second and fourth central moments of $Y^{(l,h)}$ conditional on $\mathbf{X}^{(l)}_{-P_1(\pi)}$. Notice that because $\widehat{V}_l$ is a sample variance of $Y^{(l,h)}$ conditional on $\mathbf{X}^{(l)}_{-P_1(\pi)}$, $\widehat{V}_l$ is an unbiased estimator of $M_2^{(l)} = \mathrm{Var}[Y^{(l,h)}|\mathbf{X}^{(l)}_{-P_1(\pi)}]$ conditional on $\mathbf{X}^{(l)}_{-P_1(\pi)}$ and we can use the variance expression provided by Song and Nelson [28] to derive (18). Given the budget $B$, $C = N_O N_I$ is constant. Hence, (18) can be written as

$$\frac{1}{C}\mathrm{E}[M_4^{(l)}] + \frac{1}{C}\left(-\frac{N_I-3}{N_I-1}\mathrm{E}[(M_2^{(l)})^2] + N_I\mathrm{Var}[M_2^{(l)}]\right). \qquad (19)$$

Therefore, $N_I$ that minimizes $h(N_I) = -(N_I-3)/(N_I-1)\mathrm{E}[(M_2^{(l)})^2] + N_I\mathrm{Var}[M_2^{(l)}]$ also minimizes $\mathrm{Var}[\widehat{c}(P_1(\pi))]$. Assuming $N_I$ can take real values,

$$N_I{}^\star = \operatorname*{argmax}_{N_I} h(N_I) = \begin{cases} C & \text{if } \mathrm{Var}[M_2^{(l)}] = 0, \\ \min\{1 + \sqrt{2\mathrm{E}[(M_2^{(l)})^2]/\mathrm{Var}[M_2^{(l)}]}, C\} & \text{otherwise.} \end{cases}$$

Therefore, the optimal integer-valued $N_I^{\star\star}$ that minimizes $\mathrm{Var}[\widehat{c}(P_1(\pi))]$ is

$$N_I{}^{\star\star} = \begin{cases} C & \text{if } N_I{}^\star = C, \\ \lfloor N_I{}^\star \rfloor & \text{if } h(\lfloor N_I{}^\star \rfloor) \le h(\lceil N_I{}^\star \rceil) \\ \lceil N_I{}^\star \rceil & \text{otherwise.} \end{cases}$$

Hence, $N_I{}^{\star\star}$ depends on $\mathrm{E}[(M_2^{(l)})^2]$ and $\mathrm{Var}[M_2^{(l)}]$. In other words, it depends on the functional structure of $\eta(\cdot)$ and the joint distribution $\mathbf{G}_\mathcal{K}$. For instance, assume we have two inputs $X_1$ and $X_2$ which are jointly normally distributed with 0 means, $\mathrm{Var}[X_i] = \sigma_i^2$, and $\mathrm{Corr}(X_1, X_2) = \rho$. If the model is $\eta(X_1, X_2) = X_1 + X_2$, then $M_2^{(l)} = \mathrm{Var}[X_1^{(l,h)} + X_2^{(l)}|X_2^{(l)}]$ becomes constant, which makes $\mathrm{Var}[M_2^{(l)}] = 0$. Thus, $N_I{}^{\star\star} = C$. However, if $\eta(X_1, X_2) = X_1 X_2$, then $\mathrm{Var}[M_2^{(l)}] \neq 0$ and $N_I{}^{\star\star} = 3$.

Also, here we focused on minimizing $\mathrm{Var}[\widehat{c}(P_1(\pi))]$; however, it is not guaranteed that the same setting of $N_I{}^{\star\star}$ is optimal to minimize $\mathrm{Var}[\widehat{c}(P_1(\pi) \cup \{1\})]$. Moreover, each $X_i$ may have a different $N_I{}^{\star\star}$ that minimizes $\mathrm{Var}[\widehat{Sh}_i]$.

Nevertheless, we can show that $\mathrm{Var}[\widehat{c}(P_1(\pi))]$ is smaller when $N_I = 3$ than when $N_I = 2$ regardless of $\eta(\cdot)$ as $h(3) \le h(2)$. Also, $\frac{\partial^2}{\partial {N_I}^2} h(N_I) \ge 0$ for all $N_I \ge 2$ regardless of $\eta(\cdot)$. This implies that even if ${N_I}^{\star\star} > 3$, the gain from increasing $N_I$ from 3 decreases as $N_I$ increases. Hence, we recommend $N_I = 3$ and $N_O = C/3$ to other settings. ∎

The next applies to when $k$ is large.

Claim 2. *When the permutations are randomly sampled in Algorithm* 1*, the near-optimal allocation of the computation budget is* $N_I = 3, N_O = 1$*, and* $m = C/3$ *for some constant* $C$*.*

*Proof.* If $k$ is large and therefore the permutations must be randomly sampled, then we need to take $m$ into account for the computation budget allocation. Unlike the previous case, the estimator of $Sh_1$ from Algorithm 1 is

$$\widehat{Sh}_1 = \frac{1}{m}\sum_{\ell=1}^{m}(\widehat{c}(P_1(\pi_\ell) \cup \{1\}) - \widehat{c}(P_1(\pi_\ell))),$$

where $\pi_1, \pi_2, \ldots, \pi_m$ are independent and identically distributed permutations of $k$ inputs. Therefore,

$$\mathrm{Var}[\widehat{Sh}_1] = \frac{1}{m}(\mathrm{Var}[\widehat{c}(P_1(\pi_\ell) \cup \{1\})] + \mathrm{Var}[\widehat{c}(P_1(\pi_\ell))]), \tag{20}$$

and $\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))]$ is decomposed as

$$\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))] = \mathrm{E}_{\pi_\ell}[\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))|\pi_\ell]] + \mathrm{Var}_{\pi_\ell}[\mathrm{E}[\widehat{c}(P_1(\pi_\ell))|\pi_\ell]], \tag{21}$$

where the outer expectation and variance on the right-hand side are taken with respect to the random permutation $\pi_\ell$. Notice that the conditional variance $\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))|\pi_\ell]$ is the same as the variance in (18) for the given $\pi_\ell$. Also, given $\pi_\ell$, $\widehat{c}(P_1(\pi_\ell))$ is an unbiased estimator of $c(P_1(\pi_\ell))$. Therefore, (21) becomes

$$\begin{aligned}
&\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))] \\
&= \frac{1}{N_I N_O}\left(\mathrm{E}_{\pi_\ell}\left[\mathrm{E}[M_4^{(l)}|\pi_\ell]\right] - \frac{N_I - 3}{N_I - 1}\mathrm{E}_{\pi_\ell}\left[\mathrm{E}[(M_2^{(l)})^2|\pi_\ell]\right] + N_I \mathrm{E}_{\pi_\ell}\left[\mathrm{Var}[M_2^{(l)}|\pi_\ell]\right]\right) \\
&\quad + \mathrm{Var}_{\pi_\ell}\left[c(P_1(\pi_\ell))\right] \\
&= \frac{1}{N_I N_O}\left(\mathrm{E}[M_4^{(l)}] - \frac{N_I - 3}{N_I - 1}\mathrm{E}[(M_2^{(l)})^2] + N_I \mathrm{E}_{\pi_\ell}\left[\mathrm{Var}[M_2^{(l)}|\pi_\ell]\right]\right) + \mathrm{Var}_{\pi_\ell}\left[\mathrm{E}[M_2^{(l)}|\pi_\ell]\right].
\end{aligned}$$

The last equation is from definition (7) of the cost function $c$:

$$c(P_1(\pi_\ell)) = \mathrm{E}[\mathrm{Var}[Y^{(l,h)}|\mathbf{X}^{(l)}_{-P_1(\pi_\ell)}]] = \mathrm{E}[M_2^{(l)}|\pi_\ell].$$

Letting $C = mN_O N_I$, $\frac{1}{m}\mathrm{Var}[\widehat{c}(P_1(\pi_\ell))]$ can be written as

$$\frac{1}{C}\mathrm{E}[M_4^{(l)}] - \frac{1}{C}\left(\frac{N_I - 3}{N_I - 1}\mathrm{E}[(M_2^{(l)})^2] - N_I \mathop{\mathrm{E}}_{\pi_\ell}\left[\mathrm{Var}[M_2^{(l)}|\pi_\ell]\right]\right) + \frac{N_I N_O}{C}\mathop{\mathrm{Var}}_{\pi_\ell}\left[\mathrm{E}[M_2^{(l)}|\pi_\ell]\right]. \tag{22}$$

Hence, the optimal $N_O$ to minimize (22) is 1. Similar to the previous case, we can show that (22) is smaller when $N_I = 3$ than when $N_I = 2$, which holds for any $\pi_\ell$ or any inputs other than $X_1$. Therefore, we recommend $N_I = 3, N_O = 1$, and $m = C/3$ in this case. ∎

**Appendix C. Jackson network model.** The inputs of the Jackson network model in section 5.1 are the daily order arrival rates, $\lambda_1, \lambda_2, \ldots, \lambda_6$, of six types of products. The daily order arrival rates of workstations A–F in Figure 2 are

$$\begin{aligned}
\nu_A &= \lambda_1, \\
\nu_B &= 0.4\lambda_1 + \lambda_2 + \lambda_3 + \lambda_5, \\
\nu_C &= 0.3\lambda_1 + 0.15\lambda_4 + 0.15\lambda_6, \\
\nu_D &= 0.6\lambda_1 + 0.3\lambda_4 + 0.3\lambda_6, \\
\nu_E &= \lambda_4 + \lambda_6, \\
\nu_F &= 0.85\lambda_4 + 0.85\lambda_6 + 0.3\lambda_1.
\end{aligned} \tag{23}$$

The expected job completion time of the network is

$$\eta(\lambda_1, \lambda_2, \ldots, \lambda_6) = \left\{ \sum_{j=A}^{F} \frac{\nu_j}{\mu_j - \nu_j} \right\} \times \frac{24}{\sum_{i=1}^{6} \lambda_i},$$

where $\lambda_A, \lambda_B, \ldots, \lambda_F$ are given in (23) and $\mu_A, \mu_B, \ldots, \mu_F$ are given in section 5.1. Note that we multiply by 24 to compute the job completion time in hours.

**Appendix D. Fire-spread model.** The following set of equations, based on Rothermel's fire-spread model (Rothermel [19]), are used for our experiments. The output of the model is the rate of the spread of the front point of a fire, $R$. The inputs to the model are $\delta, \sigma, h, \rho_p, m_\ell, m_d, S_T, U$, $\tan\phi$, and $P$, whose distributions and descriptions are in Table 2. Generating the random variates of these inputs is described in section 5.2. All the other variables that appear in (24)–(45) are intermediate. The equations are sorted in the order in which they should be computed.

We adopted the modifications on (28) and (36) by Albini [1] and (29), (30), and (31) by Catchpole and Catchpole [8]. Also, we conjectured (25), the relationship between the fuel loading ($w_0$) and the fuel depth ($\delta$), from the fitted sigmoidal curve in Figure 1 of Salvador et al. [25]. Note that the equations from Rothermel [19] are in imperial units, whereas the inputs in Table 2 are in metric units. The conversion factors between two systems can be found in Wagner [30].

(24) $$R = \frac{I_R \xi (1 + \phi_W + \phi_S)}{\rho_b \varepsilon Q_{ig}}$$ rate of fire-spread, ft/min

where

(25) $$w_0 = 1/(1 + \exp((15 - \delta)/2))/4.8824$$ fuel loading, $\mathrm{kg\,m^{-2}}$

(26) $$\Gamma_{max} = \sigma^{1.5}(495 + 0.0594\sigma^{1.5})^{-1}$$ maximum reaction velocity, $\mathrm{min^{-1}}$

(27) $$\beta_{op} = 3.348\sigma^{-0.8189}$$ optimum packing ratio

(28) $$A = 133\sigma^{-0.8189}$$

(29) $$\theta^\star = (301.4 - 305.87(m_l - m_d) + 2260 m_d)/(2260 m_l)$$

(30) $$\theta = \min(1, \max(\theta^\star, 0))$$

(31) $$\mu_M = \exp[-7.3 P m_d - (7.3\theta + 2.13)(1 - P) m_l]$$ moisture damping coefficient

(32) $\mu_S = 0.174 S_T^{-0.19}$ mineral damping coefficient

(33) $C = 7.47 \exp(-0.133\sigma^{0.55})$

(34) $B = 0.02526\sigma^{0.54}$

(35) $E = 0.715 \exp(-3.59 \times 10^{-4}\sigma)$

(36) $w_n = w_0(1 - S_T)$ net fuel loading, $\mathrm{lb/ft^2}$

(37) $\rho_b = w_0/\delta$ ovendry bulk density, $\mathrm{lb/ft^3}$

(38) $\varepsilon = \exp(-138/\sigma)$ effective heating number

(39) $Q_{ig} = 130.87 + 1054.43 m_d$ heat of preignition, Btu/lb

(40) $\beta = \rho_b/\rho_p$ packing ratio

(41) $\Gamma = \Gamma_{max}(\beta/\beta_{op}) A \exp[A(1 - \beta/\beta_{op})]$ optimum reaction velocity, $\mathrm{min^{-1}}$

(42) $\xi = (192 + 0.2595\sigma)^{-1} \exp[(0.792 + 0.681\sigma^{0.5})(\beta + 0.1)]$ propagating flux ratio

(43) $\phi_W = CU^B (\beta/\beta_{op})^{-E}$ wind coefficient

(44) $\phi_S = 5.275\beta^{-0.3}(\tan\phi)^2$ slope factor

(45) $I_R = \Gamma w_n h \mu_M \mu_S$ reaction intensity, $\mathrm{Btu/ft^2 min}$

## REFERENCES


[1] F. A. Albini, *Estimating Wildfire Behavior and Effects*, General technical report INT, Intermountain Forest and Range Experiment Station, Department of Agriculture, Forest Service, 1976.

[2] S. Asmussen and P. Glynn, *Stochastic Simulation: Algorithms and Analysis: Algorithms and Analysis*, Stoch. Model. Appl. Probab. 57, Springer, Berlin, 2007.

[3] L. Breiman, *Random forests*, Machine Learning, 45 (2001), pp. 5–32.

[4] M. C. Cario and B. L. Nelson, *Autoregressive to anything: Time-series input processes for simulation*, Oper. Res. Lett., 19 (1996), pp. 51–58.

[5] M. C. Cario and B. L. Nelson, *Modeling and Generating Random Vectors with Arbitrary Marginal Distributions and Correlation Matrix*, Tech. report, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 1997.

[6] F. Carreras and J. M. Giménez, *Power and potential maps induced by any semivalue: Some algebraic properties and computation by multilinear extensions*, European J. Oper. Res., 211 (2011), pp. 148–159.

[7] J. Castro, D. Gómez, and J. A. T. Cazorla, *Polynomial calculation of the Shapley value based on sampling*, Comput. Oper. Res., 36 (2009), pp. 1726–1730.

[8] E. Catchpole and W. Catchpole, *Modelling moisture damping for fire spread in a mixture of live and dead fuels*, Internat. J. Wildland Fire, 1 (1991), pp. 101–106.

[9] G. Chastaing, F. Gamboa, and C. Prieur, *Generalized Hoeffding-Sobol decomposition for dependent variables: Application to sensitivity analysis*, Electron. J. Stat., 6 (2012), pp. 2420–2448, doi:10.1214/12-EJS749.

[10] R. E. Clark, A. S. Hope, S. Tarantola, D. Gatelli, P. E. Dennison, and M. A. Moritz, *Sensitivity analysis of a fire spread model in a chaparral landscape*, Fire Ecology, 4 (2008), pp. 1–13.

[11] P. Dubey, A. Neyman, and R. J. Weber, *Value theory without efficiency*, Math. Oper. Res., 6 (1981), pp. 122–128, doi:10.1287/moor.6.1.122.

[12] T. Homma and A. Saltelli, *Importance measures in global sensitivity analysis of nonlinear models*, Reliability Engineering System Safety, 52 (1996), pp. 1–17.

[13] G. Hooker, *Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables*, J. Comput. and Graph. Statist., 16 (2007), pp. 709–732.

[14] J. R. Jackson, *Jobshop-like queueing systems*, Management Sci., 50 (2004), pp. 1796–1802.

[15] J. P. C. Kleijnen, *Factor screening in simulation experiments: Review of sequential bifurcation*, in Advancing the Frontiers of Simulation, C. Alexopoulos, D. Goldsman, and J. R. Wilson, eds., Internat. Ser. Oper. Res. Management Sci. 133, Springer, New York, 2009, pp. 153–167.
[16] R. Nelsen, *An Introduction to Copulas*, Lecture Notes in Statist., Springer, New York, 2013.
[17] J. E. Oakley and A. O'Hagan, *Probabilistic sensitivity analysis of complex models: A Bayesian approach*, J. Roy. Statist. Soc., Ser. B, 66 (2004), pp. 751–769.
[18] A. B. Owen, *Sobol' indices and Shapley value*, SIAM/ASA J. Uncertain. Quantif., 2 (2014), pp. 245–251, doi:10.1137/130936233.
[19] R. C. Rothermel, *A Mathematical Model for Predicting Fire Spread in Wildland Fuels*, USDA Forest Service Research Paper INT-155, Intermountain Forest and Range Experiment Station, Department of Agriculture, 1972.
[20] A. Saltelli, *Making best use of model evaluations to compute sensitivity indices*, Comput. Phys. Comm., 145 (2002), pp. 280–297.
[21] A. Saltelli, K. Chan, and E. Scott, *Sensitivity Analysis: Gauging the Worth of Scientific Models*, Wiley Ser. Probab. Stat., Wiley, New York, 2000.
[22] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global Sensitivity Analysis: The Primer*, Wiley, New York, 2008.
[23] A. Saltelli and S. Tarantola, *On the relative importance of input factors in mathematical models: Safety assessment for nuclear waste disposal*, J. Amer. Statist. Assoc., 97 (2002), pp. pp. 702–709, http://www.jstor.org/stable/3085706.
[24] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, Wiley, New York, 2004.
[25] R. Salvador, J. Piol, S. Tarantola, and E. Pla, *Global sensitivity analysis and scale effects of a fire propagation model used over Mediterranean shrublands*, Ecological Model., 136 (2001), pp. 175–189.
[26] L. Shapley, *A value for N-person games.*, Defense Technical Information Center, 1952.
[27] I. M. Sobol', *Sensitivity estimates for nonlinear mathematical models*, Math. Model. Comput. Experiment, 1 (1993), pp. 407–414.
[28] E. Song and B. L. Nelson, *Quickly assessing contributions to input uncertainty*, IIE Trans., 47 (2015), pp. 893–909, doi:10.1080/0740817X.2014.980869.
[29] Y. Sun, D. W. Apley, and J. Staum, *Efficient nested simulation for estimating the variance of a conditional expectation.*, Oper. Res., 59 (2011), pp. 998–1007.
[30] C. E. Van Wagner, *Metric Units and Conversion Factors for Forest Fire Quantities*, Tech. report, Petawawa Forest Experiment Station, 1978.