Why Shapley Value and Its Generalizations Are Effective in Economics and Finance, Machine Learning, and Systems Engineering

Miroslav Svitek, Niklas Winnewisser, Michael Beer, Olga Kosheleva, and
Vladik Kreinovich

Abstract In many practical situations, it is necessary to fairly divide the joint gain between the contributors. In the 1950s, the Nobelist Lloyd Shapley showed that under some reasonable conditions, there is only one way to make this division. The resulting Shapley value is now actively used in situations that go beyond economics and finance – and in which Shapley's conditions are not always satisfied: in machine learning, in systems engineering, etc. In this paper, we explain why Shapley value can be applied to such situations, and how can we generalize Shapley value to make it even more adequate for these new applications.

1 Formulation of the problem

Practical problem with which all this started. This story starts with an important practical problem: we have a group of *n* people working together on a project. This

---

Miroslav Svitek
Faculty of Transportation Sciences, Czech Technical University in Prague, Konviktska 20,
110 00 Praha 1, Czech Republic, e-mail: miroslav.svitek@cvut.cz

Niklas Winnewisser
Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34,
30167 Hannover, Germany, e-mail: winnewisser@irz.uni-hannover.de

Michael Beer
Institute for Risk and Reliability, Leibniz University Hannover, Callinstr. 34,
30167 Hannover, Germany, e-mail: beer@irz.uni-hannover.de

Olga Kosheleva
Department of Teacher Education, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: olgak@utep.edu

Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: vladik@utep.edu

project is a success. What is the fair way to divide the resulting gain between the participants?

What information can we use to make a fair division? To decide who contributed more to the success, a reasonable idea is to take into account, for all possible subsets S of the set $N \stackrel{\text{def}}{=} \{1,2,\ldots,n\}$ of all the participants, the amount $v(S)$ that they could gain if they worked on their own, without others. Based on this function v – that transforms a subset S into a real number $v(S)$ – we need to decide how to distribute the amount $v(N)$ that they earned together between the participants, i.e., which amount $x_i(v)$ to allocate to each participant so that $x_1(v)+\ldots+x_n(v)=v(N)$.

For example, if Participant i did not contribute to the gain, i.e., if we have

$$v(S\cup\{i\})=v(S)$$

for all S, then clearly this participant should not get anything: we should have

$$x_i(v)=0.$$

What do we mean by fair? The solution to this practical problem was proposed, in the early 1950s, by the future Nobelist Lloyd Shapley; see, e.g., [5, 6, 7, 8, 9, 10]. He showed that under some reasonable conditions – including the above condition that a non-contributor should not get anything – there is only one way to solve this problem. This way is now called the *Shapley value*. Here is how Shapley formalized fairness.

First, fairness means that the result should not depend on how we order the participants. In precise terms, if we perform any permutation $\pi : N \mapsto N$, then for the resulting function $v'(S) \stackrel{\text{def}}{=} v(\pi(S))$, the gains $x_i(v')$ should be the same for the same participants, i.e., we should have $x'_{\pi(i)}(v')=x_i(v)$.

Second, it means that the distribution should only depend on this particular situation and not on anything else. In other words, if the same group participates in two projects, with gains $u(S)$ and $v(S)$, then, whether we view them as two project, or a single project with gains $w(u)=u(S)+v(S)$, each participant's overall gain should be the same: $x_i(u+v)=x_i(u)+x_i(v)$. This condition is called *additivity*. In particular, when we combine identical situations, we get $x_i(m\cdot v)=m\cdot x_i(v)$ for all m.

Final formula for Shapley value. Shapley has shown that these two conditions – plus the condition that a non-contributor should not gain anything – uniquely determine the distribution of gain. The resulting formula is

$$x_i(v)=\sum_{S:\, i\notin S}\frac{|S|!\cdot(n-|S|)!}{n!}\cdot(v(S\cup\{i\})-v(S)), \tag{1}$$

where $|S|$ denotes the number of elements in a set S, and $n!$ is the factorial:

$$n! \stackrel{\text{def}}{=} 1 \cdot 2 \cdot \ldots \cdot n.$$

Shapley value is now effectively used in other applications, but why? In the last decade, Shapley value has been effectively used in many other applications, e.g., in machine learning, where it is used to describe which features are most important in a classification problem. Why Shapley value successfully works in such applications is a big mystery, because, e.g., in machine learning, one of the possible applications is to the case when $v(S)$ is the frequency of cases in which, if we only use features from the set S, we still get a correct classification.

In this application, adding probabilities does not make any sense, so we do not have additivity condition – one of the important conditions that lead to Shapley value. However, Shapley value seems to work well in this application.

What we do in this paper. In this paper, we explain:

- why the original Shapley value works well in these applications, and
- how to modify Shapley value so that it will work even better in these new applications.

2 Analysis of the problem – and the resulting explanation of why Shapley value can be applied to machine learning etc.

What are the new problems to which Shapley value is now applied: analysis. Let us recall different applications to which Shapley value is now applied.

Case of machine learning. Let us start with the application to machine learning. In this application, we have n features based on which we have a reasonably good classification – e.g., classification of images into images or cats and dogs, or, to give a more serious example, a classification of X-ray breast images into cancerous and benign. In many such cases, the full use of all these features requires a lot of processing, more than is possible to implement in each medical facility. So, a natural idea is to try to use fewer features.

To decide which features to use, ideally, we need to know, for each set S of features, what will be the probability $p(S)$ that using only these features will still lead to a correct classification. For n features, there are 2^n possible sets $S \subseteq \{1, \ldots, n\}$. When n is large, the number 2^n is astronomical – e.g., for $n = 300$, the number of possible subsets is larger than the number of particles in the Universe. We therefore cannot test every set S. Instead, a reasonable idea is to have some few-parametric approximation of the actual function $p(S)$. Specifically, what people do in machine learning applications is use the following linear approximation:

$$p(S) \approx v_{\approx}(S) \stackrel{\text{def}}{=} c_0 + \sum_{i:\, i \in S} c_i, \tag{2}$$

for some coefficients c_i – and the coefficients c_i are calculated by using the Shapley value formula.

Other applications – possible and actual. There are many other problems in which it is desirable to find a few-parameter approximation to a function of sets:

- When we have n companies that are potentially interested in merging, it is desirable to predict the gain $v(S)$ (or loss) if companies from a set $S \subseteq \{1,\ldots,n\}$ merge together; see, e.g., [4, 12, 13, 14].
- When we have n researchers, it is desirable to predict, for each subset S, what will be the productivity gain when researchers from a set S start actively collaborating. Sometimes, collaboration works, sometimes, it does not, it would be nice to be able to predict when it will work; see, e.g., [4, 12, 13, 14].
- When we have a system consisting of n components, and these components are not 100% reliable, it is desirable to predict the system's productivity $v(S)$ in situations when only components from the set S remain working; see, e.g., [16].

How is the success of Shapley value is explained now. A possible explanation for the success of Shapley value in situations like machine learning (when additivity does not seem to be applicable), was provided in [2] (see also [1]). Specifically, if we use the usual least squares approach (see, e.g., [11]) to find the coefficients that best fit the approximate formula (2), then we get linear expressions for the coefficients Δv_i in terms of the values $v(S)$. In particular, one of the options is the Shapley value, but more general expressions are also possible – depending on what standard deviations we use for different values of $|S|$.

Need to go beyond linear approximations. In many practical cases, a linear approximation (2) makes perfect sense. However, when we are interested in studying potential mergers or researcher collaboration, the whole purpose is to make sure that the joint effect is larger than the sum of the original contributions. To capture this additional joint effect, we need to go beyond the linear approximation.

How can we go beyond linear approximations? To answer this question, let us reformulate the right-hand side of the expression (2) in the following equivalent form:

$$v \approx (S) = c_0 + \sum_{i=1}^{n} c_i \cdot m_i(S), \tag{3}$$

where $m_i(S) = 1$ if $i \in S$ and $m_i(S) = 0$ otherwise. In this form, this is clear that the resulting approximation is a linear function of the variables $m_i(S)$.

What can we do if we have an approximate linear dependence, and we want to come up with a more accurate approximation? Such a situation is typical in physics; see, e.g., [3, 15]. A usual physics approach is to take into account that most real-world dependencies $f(x_1,\ldots,x_n)$ are smooth, they can be expanded in Taylor series

$$f(x_1,\ldots,x_n) = c_0 + \sum_{i_1=1}^{n} c_{i_1} \cdot x_{i_1} + \sum_{i_1=1}^{n} \sum_{i_2=1}^{n} c_{i_1 i_2} \cdot x_{i_1} \cdot x_{i_2} + \ldots \tag{4}$$

In general, the linear terms are the largest, quadratic terms are second largest, etc. So, if we are not satisfied with an accuracy of the linear approximation, a natural idea is to also take into account quadratic terms. If this will not lead to a sufficiently accurate approximation, we can take cubic terms into account, etc. In general, we thus get the following approximate formula

$$f_{\approx}(x_1,\ldots,x_n) = c_0 + \sum_{i_1=1}^{n} c_{i_1}\cdot x_{i_1} + \sum_{i_1=1}^{n}\sum_{i_2=1}^{n} c_{i_1 i_2}\cdot x_{i_1}\cdot x_{i_2} + \ldots + \sum_{i_1=1}^{n}\ldots\sum_{i_k=1}^{n} c_{i_1\ldots i_k}\cdot x_{i_1}\cdot\ldots\cdot x_{i_k}. \quad (5)$$

It is reasonable to apply the same approach to our case as well – and this was indeed proposed, e.g., in [4, 12, 13, 14, 16]. The only difference from the general formulas (4)–(5) is that in our case, each value $m_i(S)$ is equal either to 0 or to 1, $(m_i(S))^k = 1$ for all $k \geq 1$ and thus, it does not make sense to separately consider terms like $(m_i(S))^2$, etc. So, we arrive at the following formula:

$$v_{\approx}(S) = c_0 + \sum_{i_1} c_{i_1}\cdot m_{i_1}(S) + \sum_{i_1<i_2} c_{i_1 i_2}\cdot m_{i_1}(S)\cdot m_{i_2}(S) + \ldots + \sum_{i_1<i_2<\ldots<i_k} c_{i_1\ldots i_k}\cdot m_{i_1}(S)\cdot\ldots\cdot m_{i_k}(S). \quad (6)$$

Actually, in many cases, there is additivity. In cases of gain – like companies merger or researcher collaboration – we do have the same additivity idea as in the original applications of Shapley value.

In the machine learning case, it does not make sense to add probabilities, but we *can* add gain coming from the increased probability of success. This gain is proportional to probability of success, so additivity of gain naturally translates into additivity of probabilities. *This, by the way, explains why the Shapley value is so effective in machine learning.*

Similarly, in the systems engineering case, we can measure the results of only-some-components-functioning in terms of productivity; in this case, we also have a natural idea of adding productivity of both cases – so we can also formulate a natural additivity condition.

The desired approximation should be exact in some cases. For the usual Shapley value, the linear approximation is exact for the case when S is the empty set: if no one does anything, there is clearly no gain. For the proposed Taylor-series-type generalization, we have more parameters, so we can add more sets for which this approximation is exact. We can start with this exactness requirement for sets S consisting of a single element, then formulate a similar requirement for sets consisting of two elements, etc.

In the next section, we will show that with this additional requirement, there is a unique Taylor-type expression that is additive and symmetric.

3 Our main result

Definition. *Let $n \geq 2$ and $k \geq 1$ be integers.*

- *By a* set function *we mean a function that assigns, to each set $S \subseteq N \stackrel{\text{def}}{=} \{1,\ldots,n\}$, a number $v(S)$ so that $v(\emptyset) = 0$.*
- *By a k-*approximation*, we mean an expression of the type*

$$v_{\approx}(S,v) = c_0(v) + \sum_{i_1} c_{i_1}(v) \cdot m_{i_1}(S) + \sum_{i_1<i_2} c_{i_1 i_2}(v) \cdot m_{i_1}(S) \cdot m_{i_2}(S) + \ldots +$$

$$\sum_{i_1<i_2<\ldots<i_k} c_{i_1\ldots i_k}(v) \cdot m_{i_1}(S) \cdot \ldots \cdot m_{i_k}(S), \tag{7}$$

for some continuous functions $c_{i_1\ldots}(S)$.
- *We say that the k-approximation is* dummy-fair *if whenever for some set A, we have $v(S \cup A) = v(S)$ for all S, we should get $v_{\approx}(S \cup A, v) = v_{\approx}(S,v)$ for all S. In particular, for $S = \emptyset$, we should get $v_{\approx}(A,v) = v_{\approx}(\emptyset,v)$.*
- *We say that the k-approximation is* symmetric *if for every permutation $\pi : N \mapsto N$, we have $v_{\approx}(\pi(S),\pi(v)) = \pi(v_{\approx}(S,v))$, where $(\pi(v))(S) \stackrel{\text{def}}{=} v(\pi(S))$.*
- *We say that the k-approximation is* additive *for every two set functions u and v, we have $v_{\approx}(S,u+v) = v_{\approx}(S,u) + v_{\approx}(x,v)$.*
- *We say that the k-approximation is* exact for small sets *if $v_{\approx}(S,v) = v(S)$ for all sets S for which $|S| < k$.*

Proposition. *For each n and k, there exists exactly one k-approximation which is dummy-fair, symmetric, additive, and exact for small sets:*

$$v_{\approx}(S,v) = \sum_{T:\ |T| \leq k} \left(\frac{w_v(T)}{|T|} \cdot \prod_{i:\ i \in T} m_i(S) \right) +$$

$$\sum_{T:\ |T|>k} \left(\frac{k! \cdot (|T|-k)!}{|T|!} \cdot \sum_{U:\ U \subset T \,\&\, |U|=k} \left(\frac{w_v(U)}{|U|} \cdot \prod_{i:\ i \in U} m_i(S) \right) \right), \tag{8}$$

where

$$w_v(T) \stackrel{\text{def}}{=} \sum_{S:\ S \subseteq T} (-1)^{|T|-|S|} \cdot v(T). \tag{9}$$

Comment. For $k = 1$, we get a linear expression for which the coefficients c_i are exactly the Shapley values.

Example beyond Shapley value. If $n = 3$, and we get some result only when at least 2 people work together, i.e., if $v(S) = 0$ for all 1-element sets and $v(S) = 1$ for all 2-element sets and for the whole set N, then:

- For $k = 1$, we get a linear approximation based on the Shapley values:

$$v_{\approx}(S) = \frac{1}{3} \cdot m_1(S) + \frac{1}{3} \cdot m_2(S) + \frac{1}{3} \cdot m_3(S). \tag{10}$$

This approximation is exact only for the empty set $S = \emptyset$.

- For $k = 2$, we get the following quadratic approximation:

$$v_{\approx}(S) = \frac{1}{3} \cdot m_1(S) \cdot m_2(S) + \frac{1}{3} \cdot m_1(S) \cdot m_3(S) + \frac{1}{3} \cdot m_2(S) \cdot m_3(S). \tag{11}$$

This approximation is exact for all 1-element sets S.

- For $k = 3$, we get the following cubic approximation:

$$v_{\approx}(S) = m_1(S) \cdot m_2(S) \cdot m_3(S). \tag{12}$$

This approximation is exact for all 2-elements sets S – and, it so happens that it is also exact for the whole 3-element set $N = \{1, 2, 3\}$.

Proof.

1°. First, one can see that additivity implies that each coefficient function $c_{i_1 \ldots i_\ell}(v)$ is additive, i.e., that for each such coefficient function, we have

$$c_{i_1 \ldots i_\ell}(u + v) = c_{i_1 \ldots i_\ell}(u) + c_{i_1 \ldots i_\ell}(v). \tag{13}$$

In particular, if we add m identical set functions, we get

$$c_{i_1 \ldots i_\ell}(m \cdot v) = m \cdot c_{i_1 \ldots i_\ell}(v). \tag{14}$$

In particular, for any integer q, for the set function v/q, we get

$$c_{i_1 \ldots i_\ell}(v) = q \cdot c_{i_1 \ldots i_\ell}\left(\frac{v}{q}\right) \tag{15}$$

hence

$$c_{i_1 \ldots i_\ell}\left(\frac{1}{q} \cdot v\right) = \frac{1}{q} \cdot c_{i_1 \ldots i_\ell}(v). \tag{16}$$

By applying the formula (14), we conclude that for every positive rational number m/q, we have

$$c_{i_1 \ldots i_\ell}\left(\frac{m}{q} \cdot v\right) = m \cdot c_{i_1 \ldots i_\ell}\left(\frac{1}{q} \cdot v\right).$$

So, due to (16), we get:

$$c_{i_1 \ldots i_\ell}\left(\frac{m}{q} \cdot v\right) = \frac{m}{q} \cdot c_{i_1 \ldots i_\ell}(v). \tag{17}$$

Since the function $c_{i_1 \ldots i_\ell}(v)$ is continuous, and every non-negative real number a can be represented as a limit of positive rational numbers, we conclude that, in the limit when $m/q \to a$, the same equality (17) holds for every non-negative real number a:

$$c_{i_1\dots i_\ell}(a\cdot v)=a\cdot c_{i_1\dots i_\ell}(v). \tag{18}$$

Together with additivity, this means that for all possible linear combination of set functions with non-negative coefficients, we have

$$c_{i_1\dots i_\ell}(a_1\cdot v_1+\dots+a_t\cdot v_t)=a_1\cdot c_{i_1\dots i_\ell}(v_1)+\dots+a_t\cdot c_{i_1\dots i_\ell}(v_t). \tag{19}$$

By combining these equalities for all the coefficient functions $c_{i_1\dots}(v)$, we conclude that

$$v_{\approx}(S,a_1\cdot v_1+\dots+a_t\cdot v_t)=a_1\cdot v_{\approx}(S,v_1)+\dots+a_t\cdot v_{\approx}(S,v_t). \tag{20}$$

2°. Let us show that the equality (20) holds also when some coefficients a_i are negative – but the expression

$$v\stackrel{\text{def}}{=}a_1\cdot v_1+\dots+a_t\cdot v_t \tag{21}$$

is still positive and thus, makes sense. Indeed, in this case, if, in the equality (21), we move all the terms for which a_j is negative to the left-hand side, we get the following equality:

$$v+\sum_{j:\,a_j<0}|a_j|\cdot v_j=\sum_{j:\,a_j\ge 0}a_j\cdot v_j. \tag{22}$$

On both sides, the coefficients are non-negative, so we can apply the formula (20) to both sides and get the following equality:

$$v_{\approx}(S,v)+\sum_{j:\,a_j<0}|a_j|\cdot v_{\approx}(S,v_j)=\sum_{j:\,a_j\ge 0}a_j\cdot v_{\approx}(S,v_j) \tag{23}$$

Now, we can move the terms corresponding to $a_j<0$ back into the right-hand side and get

$$v_{\approx}(S,v)=\sum_j a_j\cdot v_{\approx}(S,v_j), \tag{24}$$

i.e., indeed, the formula (20) holds for all possible coefficients a_j.

3°. To complete the proof, we will use th fact – that is used in the original Shapley's proof – that for every set function $v(S)$, we have:

$$v(S)=\sum_T w_v(T)\cdot v_T(S), \tag{25}$$

where the sum is over all sets T, and $v_T(S)$ denotes the following set function:

- we have $v_T(S)=1$ when $T\subset S$ and
- we have $v_T(S)=0$ otherwise.

In other words, the set function $v_T(S)$ means the following:

- if all the participants from the set T get together, then they gain 1 unit – irrespective of what others do;
- otherwise, if not all participants from the set T get together, no one gains anything.

Because we proved that formula (20) holds for all linear combinations, with coefficients of any sign, we can conclude that

$$v_{\approx}(S,v) = \sum_{T} w_v(T) \cdot v_{\approx}(S,v_T). \tag{26}$$

So, to find the desired value $v_{\approx}(S,v)$, it is sufficient to describe the values $v_{\approx}(S,v_T)$ corresponding to all possible sets T.

4°. For each function v_T, participants outside the set T do not affect the value of the set function, and thus – due to dummy-fairness, adding these participants should not change the approximate value $v_{\approx}$:

$$v_{\approx}(S,v) = v_{\approx}(S \cap T, v). \tag{27}$$

Thus, it is sufficient to consider the values $v_{\approx}(S,v)$ for subsets S of the set T.

By definition of the set function v_T, we have $v_T(S) = 0$ for all S for which $|S| < |T|$. When $|T| \le k$, then, due to the exactness requirement, this means that we should also have $v_{\approx}(S,v) = 0$ for all S. So, we cannot have any products of fewer than $|T|$ values $m_i(S)$. So, the only possible term is the product of all $|T|$ values $m_i(S)$ corresponding to $i \in T$.

When $|T| > k$, we should also have $v_{\approx}(S,v) = 0$ for all sets S with $|S| < k$. Thus, we also cannot have any products with fewer than k values. So the only remaining option is to have products of k values. The set function $v_T(S)$ does not change if we perform any permutation of the elements of the set T. Thus, due to symmetry, once we have one such product, we should have, with the same coefficient, *all* such products corresponding to all possible subsets U of size k. From the fact that the sum of all these terms should be equal to 1, we deduce the coefficient at each such product – it is 1 divided by the number of such subsets U.

Substituting the resulting expressions for $v_{\approx}(S,v_T)$ into the formula (26), we get the desired expression (8). The proposition is proven.

Acknowledgments

This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), HRD-1834620 and HRD-2034030 (CAHSI Includes), EAR-2225395 (Center for Collective Impact in Earthquake Science C-CIES), and by the AT&T Fellowship in Information Technology.

It was also supported by a grant from the Hungarian National Research, Development and Innovation Office (NRDI), by the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany, by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Focus Program SPP 100+ 2388, Grant Nr. 501624329, and the European Union under the project ROBOPROX (reg. no. CZ.02.01.01/00/22 008/0004590).

References

1. K. Autchariyapanikul, O. Kosheleva, and V. Kreinovich, “Shapley value under interval uncertainty and partial information”, In: V. Kreinovich, W. Yamaka, and S. Leurcharusmee (eds.), *Data Science for Econometrics and Related Topics*, Springer, Cham, Switzerland, to appear.
2. L. Bokati, O. Kosheleva, V. Kreinovich, and N. N. Thach, “Why shapley value and its variants are useful in machine learning (and in other applications)”, In: V. Kreinovich, S. Sriboonchitta, and W. Yamaka (eds.), *Machine Learning for Econometrics and Related Topics*, Springer, Cham, Switzerland, 2024, pp. 169–174.
3. R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.
4. O. Galindo, M. Svitek, and V. Kreinovich, “Quantum (and more general) models of research collaboration”, *Asian Journal of Economics and Banking*, 2020, Vol. 4, No. 1, pp. 77–86.
5. R. D. Luce and R. Raiffa, *Games and Decisions: Introduction and Critical Survey*, Dover, New York, 1989.
6. R. B. Myerson, *Game Theory: Analysis of Conflict*, Harvard University Press, Cambridge, Massachisetts, 1997.
7. G. Owen, *Game Theory*, Emerald Publishers, Bingley, UK, 2013.
8. A. E. Roth (ed.), *The Shapley Value: Essays in Honor of Lloyd S. Shapley*, Cambridge University Press, Cambridge, 1988. doi:10.1017/CBO9780511528446, ISBN 0-521-36177-X.
9. L. S. Shapley, *Notes on the n-Person Game – II: The Value of an n-Person Game*, Research Memorandum RM-670, RAND Corporation, Santa Monica, California, USA, 1951.
10. L. S. Shapley, “Value for *n*-person Games”, In: H. W. Kuhn and A. W. Tucker (eds.), *Contributions to the Theory of Games*, Annals of Mathematical Studies, Vol. 28, Princeton University Press, Princeton, New Jersey, USA, 1953, pp. 307–317, doi:10.1515/9781400881970-018. ISBN 9781400881970.
11. D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.
12. M. Svitek, O. Kosheleva, and V. Kreinovich, “When is it beneficial to merge two companies? When is it beneficial to start a research collaboration?”, In: M. Ceberio and V. Kreinovich (eds.), *Uncertainty, Constraints, AI, and Decision Making*, Springer, Cham, Switzerland, to appear.
13. M. Svitek, O. Kosheleva, V. Kreinovich, and T. N. Nguyen, “Why quantum (wave probability) models are a good description of many non-quantum complex systems, and how to go beyond quantum models”, In: V. Kreinovich, N. N. Thach, N. D. Trung, and D. V. Thanh (eds.), *Beyond Traditional Probabilistic Methods in Economics*, Springer, Cham, Switzerland, 2019, pp. 168–175.
14. M. Svitek, O. Kosheleva, V. Kreinovich, and N. H. Phuong, “To which interdisciplinary research collaborations should we pay more attention?”, In: N. H. Phuong, N. T. H. Chau, and V. Kreinovich (eds.), *Explainable Artificial Intelligence and Other Soft Computing Techniques: Biomedical and Related Applications*, Springer, Cham, Switzerland, to appear.
15. K. S. Thorne and R. D. Blandford, *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*, Princeton University Press, Princeton, New Jersey, 2021.

16. N. Winnewisser, M. Beer, O. Kosheleva, and V. Kreinovich, In: V.-N. Huynh, K. Honda, B. H. Le, M. Inuiguchi, and H. T. Huynh (eds.), "How Shapley Value and Its Generalizations Can Help in the Analysis of Complex Engineering Systems and What Next", *Integrated Uncertainty in Knowledge Modelling and Decision Making. Proceedings of the 11th International Symposium, IUKM 2025, Ho Chi Minh City, Vietnam, March 17–19, 2025*, Part II, Springer Lecture Notes in Computer Science, 2025, Vol. 15586, pp. 85–97, https://doi.org/10.1007/978-981-96-4603-6_8