An Improved Tool for Distribution Envelope Determination, a Technique for Interval-Based, Verified Arithmetic on Random Variables

Daniel Berleant, Lizhi Xie, Jianzhong Zhang,
and Gerry Sheblé
Department of Electrical and Computer Engineering
Iowa State University, Ames, Iowa 50011, USA
berleant@iastate.edu

1 Summary

When random variables possessing arbitrary distribution functions must be combined via +, −, *, /, min(), max(), etc., Monte Carlo simulation is commonly employed. However, Monte Carlo simulation assumes either independence or (less commonly) some other specific dependency relationship, among other limitations (Ferson 1996). Discretization of the distribution function followed by a numerical method is an alternative. Numerical methods can relax the requirement of Monte Carlo that the distributions have a known dependency relationship, in which case the results are typically envelope curves within which the cumulative distribution of the result must lie regardless of the dependency relationship between the operands. The operands themselves can also be expressed with envelopes in order to bound the effects of discretization of the input distributions (Berleant 1993; Williamson and Downs 1990). This paper describes Statool, a software tool that implements Distribution Envelope Determination (DEnv), a numerical algorithm for performing arithmetic on distribution function operands (Berleant and Goodman-Strauss 1998). Our previously reported tool was limited to independent random variables (Berleant and Cheng 1998), a significant limitation. Improvements to Statool are currently being driven by the needs of applications in accordance with our research strategy, which is to identify such applications and then to modify Statool as needed to support them. However, identifying good applications is itself a research topic. We are currently exploring applications to the electric power industry (Sheblé and Berleant 2002; Berleant et al. 2002), and have obtained recent results on time to completion of multiple tasks and time to failure of two components [7,8].

2 Introduction

Random variables may be combined using standard operations such as +,-,*, /, min(), and max(). When the random variable operands are assumed independent, results may be calculated using a discretized convolution approach (Ingram et al. 1968; Colombo and Jaarsma 1980; Kaplan 1981). Discretization error may be bounded by an interval based extension (Berleant 1993). We have described a tool implementing this (Berleant and Cheng 1998), however it is desirable though non-trivial to extend that work by eliminating the assumption that the random variables are independent, thereby handling the case where their dependency relationship is unknown and unspecified. In this case of unspecified dependency, obtaining bounded results requires that the entire range of possible dependency relationships be accounted for, including independence as one of the infinite number of possible dependencies. While the traditional approach of Monte Carlo simulation does not bound the range of results that are possible when dependency is unspecified (Ferson 1996), the desired bounds can be obtained with other techniques. A copula-based approach (Frank et al. 1987) which was significantly extended by Williamson and Downs (1990) and termed Probabilistic Arithmetic, has been implemented in a commercially available software system, RiskCalc (Ferson et al. 1998). DEnv (Distribution Envelope Determination) is described by Berleant and Goodman-Strauss (1998). A comparison of DEnv and Probabilistic Arithmetic reveals underlying similarities (Regan et al., submitted), as well as differences (Berleant and Goodman-Strauss 1998) that motivate its software implementation as well as continued development in other ways.

This paper reports a software implementation of DEnv (see Figures 1 and 2). This tool represents an advance over our previously developed tool, as described next.

- Calculation of $z = f(x, y)$ when x and y are not assumed independent (Berleant and Goodman-Strauss 1998) is now supported. The previously described tool assumes random variables are independent. The current tool bounds the range of results that are plausible when independence is not assumed. Figure 1 shows an example.

- Calculation of $max(x, y)$ and $min(x, y)$ for random variables x and y is now supported. This can be useful in problems like determining the time to complete two concurrent tasks, because the completion time of both is the same as the completion time of the task that finishes second, i.e., the maximum of the two individual completion times.

- Calculation of $z = f(x, y)$ in some instances where the interval expression for $f(x, y)$ leads to excess width is now supported. Although in DEnv x and y are probability distributions, DEnv reduces operations on distributions to operations on intervals, and the net effect of excess width in the interval calculated for $f(x, y)$, x and y intervals, is excessively wide envelopes derived for $f(x, y)$, where x and y are distributions. The tool

handles such expressions under the severe restriction that the function is monotonic over the box defined by the range over which distributions x and y are non-zero. While it would be desirable to incorporate more advanced techniques for reducing excess width for non-monotonic functions, even the current capability extends the state of the art for performing operations on distributions of unknown dependency, allowing evaluation of expressions such as that which produced Figure 2 without excess width in the envelopes because excess width is removed from the underlying interval evaluations of the expression.

- Calculation of cascaded operations is now supported. These are cases in which the result of one operation is used as an input to the next operation. The distributions used as inputs to an operation are discretized density functions, while the output of an operation consists of bounding envelopes which are cumulative distributions. Thus to use the output of an operation as the input to another operations requires converting a pair of bounding CDF envelopes into a discretized density function. We have done this by generalizing the histogram representation of an input to allow overlapping bars. This in turn enables conversion of the envelopes to the generalized histogram form, as will be described in the full paper. The generalized histogram form can then be used as an input to an operation the same way an ordinary histogram discretization of a density function can.

3 Algorithmic Issues

Calculation of results in the case of unspecified dependency between operands is based on a joint distribution tableau in which discretizations of each operand into intervals and associated probability masses form the marginals, and the interior cells are subject to constraints imposed by the marginals. Linear programming is called subject to these constraints, as a subroutine to find each desired point on the left and right envelopes. Only a limited number of points need to be found this way, because the discrete nature of the problem allows connecting the points safely to produce staircase-like envelopes in which each point is a bend in the staircase. While many details were covered in Berleant and Goodman-Strauss (1998), the linear programming aspects were not. Therefore we will review the DEnv algorithm in the full paper, emphasizing the linear programming aspects. Details on the algorithm as it applies to particular problems, including its linear programming aspects, may also be found in other works under review and available from the authors.

References

[1] S. Ferson, "What Monte Carlo Methods Cannot Do", *Human and Ecological Risk Assessment*, 1996, Vol. 2, pp. 990–1007.

[2] D. Berleant, "Automatically verified reasoning with both intervals and probability density functions", *Interval Comutations*, 1993, No. 2, pp. 48–70.

[3] D. Berleant and H. Cheng, "A software tool for automaticaly verified reasoning on intervals and probability distributions", *Reliable Computing*, 1998, Vol. 4, No. 1, pp. 71–82.

[4] D. Berleant and C. Goodman-Strauss, "Bounding the results of arithmetic operations on random variables of unknown dependency using intervals", *Reliable Computing*, 1998, Vol. 4, No. 2, pp. 147–165.

[5] D. Berleant, J. Zhang, R. Hu, and G. Sheblé, "Economic dispatch: applying the interval-based Distribution Envelope algorithm to an Electric Power Problem", *Validated Computing 2002*, May 23–25, Toronto.

[6] G. Sheblé and D. Berleant, "Bounding the composite value at risk for energy service company operation with DEnv, an interval-based algorithm", *Validated Computing 2002*, May 23–25, Toronto.

[7] D. Berleant, J. Zhang, and G. Sheblé, *On completion times of networks of concurrent and sequential tasks*, 2001; submitted 6/01.

[8] D. Berleant, J. Zhang, and G. Sheblé, *On bounding failure times in two-component systems*, in preparation; current manuscript available upon request.

[9] A. G. Colombo and R. J. Jaarsma, "A powerful numerical method to combine random variables", *IEEE Transactions on Reliability*, 1980, Vol. R-29, No. 2, pp. 126–129.

[10] R. A. Evans, "Bayes Paradox", *IEEE Transactions on Reliability*, 1982, Vol. R-31, No. 4, p. 321.

[11] G. E. Ingram, E. L. Welker, and C. R. Herrmann, "Designing for reliability based on probabilistic modeling using remote access computer systems", *Proc. 7th Reliabiltiy and Maintainability Conference*, American Society of Mechanical Engineers, 1968, pp. 492–500.

[12] S. Kaplan, "On the Method of Discrete Probability Distributions in Risk and Reliability Calculations, Applications to Seismic Risk Assessment", *Risk Analysis*, 1981, Vol. 1, No. 3, pp. 189–196.

[13] H. Regan, S. Ferson, and D. Berleant, *Equivalence of five methods for bounding uncertainty*, submitted.

[14] R. Williamson and T. Downs, "Probabilistic arithmetic I: numerical methods for calculating convolutions and dependency bounds", *International Journal of Approximate Reasoning*, 1990, Vol. 4, pp. 89–158.

Figure 1: Two normal distributions each with $\mu = 1$ and $\sigma = 1$ were tail-trimmed to within $[-3, 5]$ (because the tool is currently limited to numerically valued bounds). These distributions were used as input variables. Given no assumptions about their dependency relationship, staircase-shaped left and right envelopes were computed which enclose the space within which the distribution of (a sufficiently large number of) products of samples of the inputs must travel regardless of their dependency relationship. There are also three smoother curves showing the product distributions for three particular dependency relationships that allow the curves to be computed relatively easily. One of these is for independent inputs, and was computed using the Monte Carlo-generated products of 100,000 samples of the inputs. The other two are analytically derived distributions of the product assuming Pearson correlations of 1 and -1.

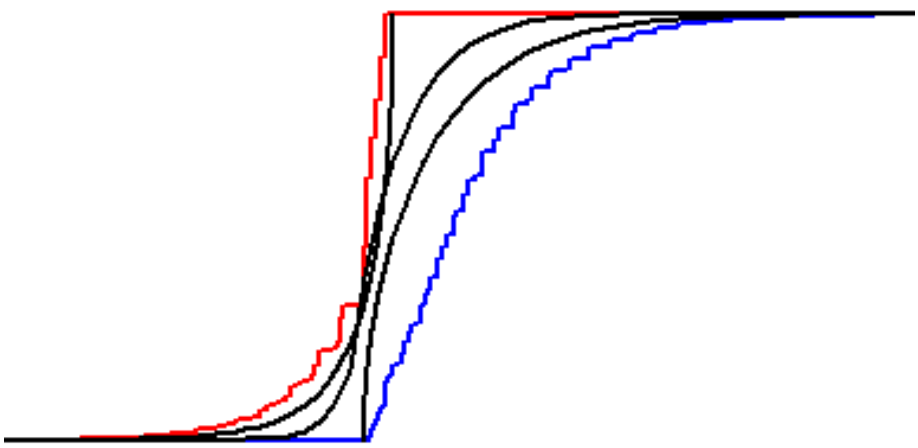

Figure 1:

Figure 2 follows on next page: X and Y are inputs. Z constitutes envelopes around the result when the dependency relationship between X and Y is unspecified, and $Z = (38 * Y - 8 * X)/(0.08 * Y + 0.048 * X)$. The cumulative forms of histogram discretizations of PDFs (X and Y) are pairs of CDF bounds that each look like two staircases in which the top bends of the lower curve touch the bottom bends of the upper curve. The cumulative form of the result does not in general obey that constraint, and hence cannot in general be displayed correctly as a histogram. It can be displayed correctly in cumulative form, as shown in the lower subwindow.

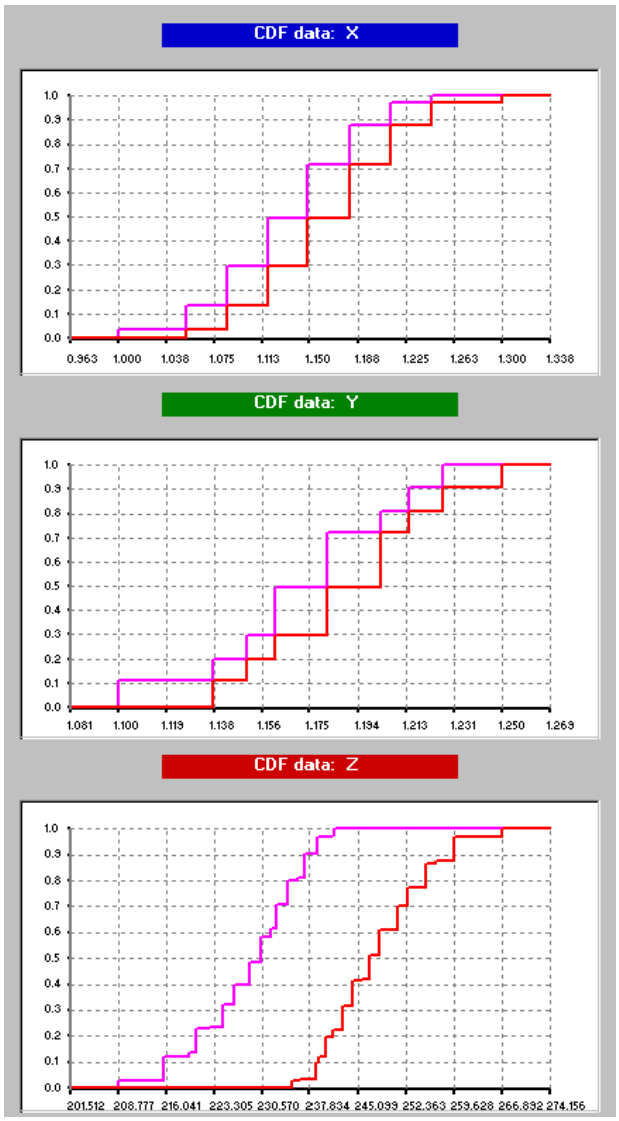
CDF data: X
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.963 1.000 1.038 1.075 1.113 1.150 1.188 1.225 1.263 1.300 1.338
CDF data: Y
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
1.081 1.100 1.119 1.138 1.156 1.175 1.194 1.213 1.231 1.250 1.269
CDF data: Z
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
201.512 208.777 216.041 223.305 230.570 237.834 245.099 252.363 259.628 266.892 274.156

Figure 2: