A Software Tool for Automatically Verified Operations on Intervals and Probability Distributions

Daniel Berleant and Hang Cheng

Abstract

We describe a software tool for performing automatically verified arithmetic operations on independent operands when the operands are intervals, or probability distribution functions, or one operand is an interval and the other is a distribution. Intervals and distributions are expressed using the same technique, so the algorithms do not need to distinguish between intervals and distributions in their operation. The tool can calculate common arithmetic operations with guaranteed results (as well as confidence limits on a distribution if the distribution is empirically estimated from samples).

A previous paper (Berleant 1993 [1]) discusses the concepts, algorithms, and related work. Here we emphasize a software tool that implements the algorithms, interacts with the user via a graphical user interface, and saves, retrieves, and prints the results of its calculations.

1 Introduction to the Representation

We use histograms to represent, correctly, both probability distribution functions (PDFs) and intervals. We take an interval to be an incompletely specified description of value such that there is a probability of one that the actual but unknown value falls within the interval. A probability distribution may be in the form of either a probability density function (PDF) or its integral, a cumulative distribution function (CDF). Next, we show how to represent both distribution functions and intervals, correctly, using the same representational technique.

1.1 Representing a Distribution Function

Histograms are a natural way for people to create and edit probability distributions, and are used for this purpose by our software tool. While a histogram discretizes the underlying PDF, in our technique this discretization introduces no error, and thereby maintains correctness. Instead, it introduces information loss. To see this, consider a histogram discretization of a PDF to be a partitioning of the PDF's domain into a set of intervals, each of which is associated with a probability mass equal to the area under the PDF across the range of that interval. The graphical depiction of a histogram is then simply a convention for describing this set of intervals and their associated masses.

A histogram graphic (Figure 1) is user friendly but has the disadvantage that a histogram bar is conventionally shown with a flat top, which might give the impression that the probability mass associated with the interval for that bar is assumed to be distributed evenly over its range. In fact, in our technique no assumption is made about how the probability mass associated with an interval is distributed over that interval, hence a given histogram bar is consistent with any such distribution, including the PDF that the histogram might have been intended to discretize but also an infinite number of others. This is explained in more detail in Berleant (1993 [1]), but the main point to consider is that this view of a histogram allows it to discretize a PDF correctly but with information loss, and displaying the bars with flat tops is merely a graphical convention. The more bars in the histogram, the less the information loss.

The software tool we report here includes the capability of allowing users to graphically edit histograms. To underline their information-losing but correctness properties, the tool also allows display of the same information in another graphical form, the integral (or cumulated probability) of a histogram. Just as a PDF can be integrated into a corresponding CDF, the histogram discretization of a PDF we have described can also be integrated and this cumulated form displayed graphically (Figure 1). The graphic for the cumulation of the histogram consists of two bounding CDFs. The higher of the two assumes the extreme case where the probability mass associated with each bar of the histogram is concentrated at the low bound of its interval. The lower of the two assumes the other extreme, where each probability mass is concentrated at the high bound of its corresponding interval. Any other of the infinite number of distributions consistent with the histogram, when integrated, results in a curve that stays within the two stair-step shaped bounding CDFs exemplified in Figure 1. The space between these bounding CDFs reflects the information loss associated with the discretization.

1.2 Representing an Interval

An interval is represented in our technique as a histogram with one bar. Since the interval representation of value does not assume any particular distribution of probability mass within the interval, the interval representation is consistent with any PDF whose value is zero outside the bounds of the interval, and with any CDF whose value is zero below the interval and one above it (Figure 2). To represent the interval using bounding CDFs, we must determine the space of CDFs consistent with the interval. The two most extreme CDFs that bound

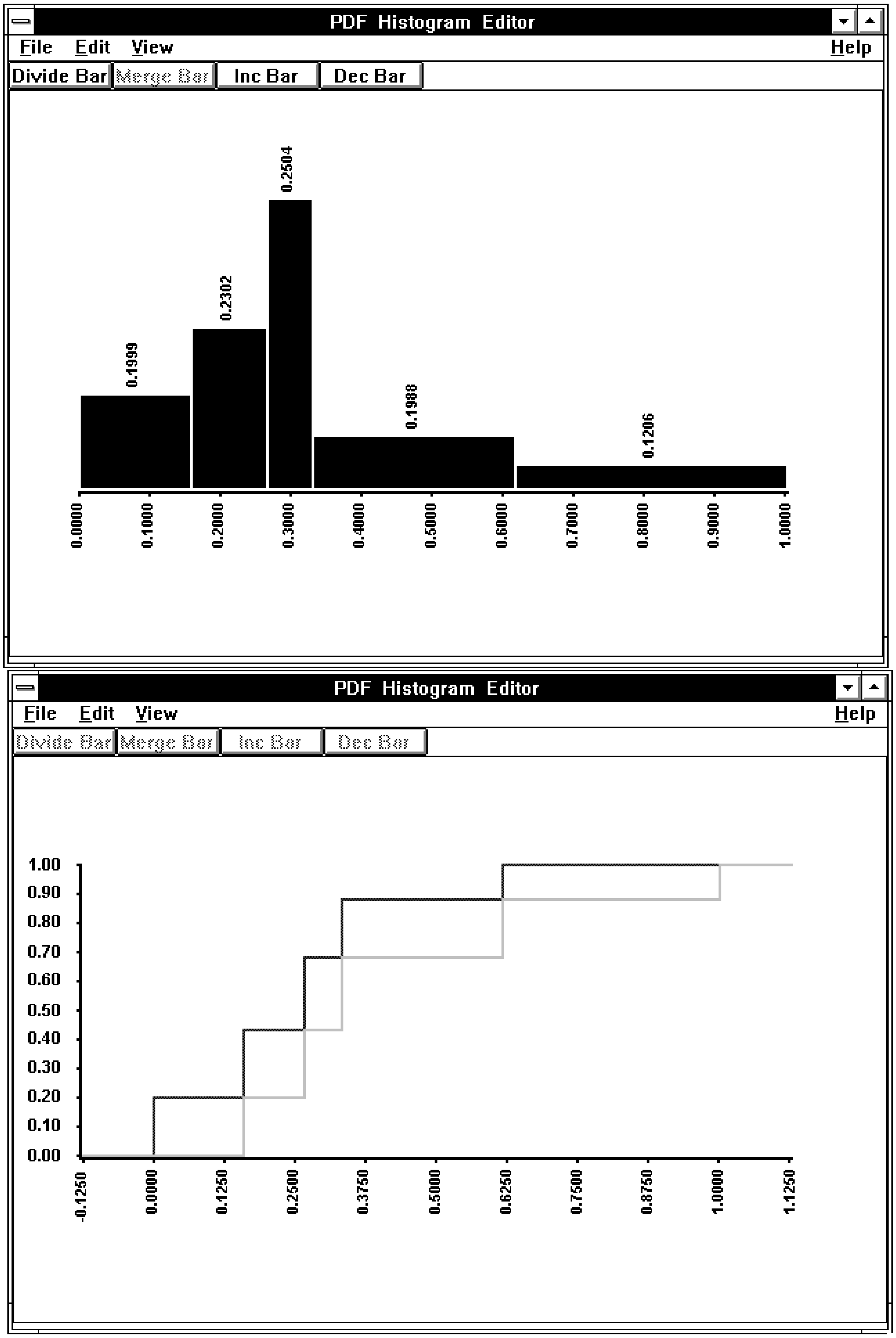

Figure 1: A histogram edited by a user and its two corresponding CDF bounds (one solid and one dotted).

this space are the one that rises faster than any other CDF consistent with the interval, and the one that rises slower than any other CDF consistent with the interval. The one that rises fastest will correspond to the PDF which is an impulse of mass 1 at the interval's low bound (that is, the variable's value is certainly equal to the interval's low bound). Likewise, the one that rises slowest will correspond to the PDF which is an impulse of mass 1 at the interval's high bound. The interval is therefore representable as a family of CDFs whose two bounding CDFs each have one "step." Figure 2 shows an interval and its representation as a bounded family of CDFs; the CDF that bounds this family from above jumps from zero to one at the interval's low bound, and CDF that bounds it from below jumps from zero to one at the interval's high bound.

2 Introduction to Arithmetic Operations

The algorithmic approach to performing an arithmetic operation on histograms is exemplified in Figure 3. An arithmetic operation produces as a result a Cartesian product consisting of a set of intervals that in general contains overlaps. Due to the overlaps this set cannot be shown directly as a histogram, but can be integrated and shown correctly as a pair of stepwise, bounding CDFs. (The software tool will however allow a result to be displayed as an *approximating* histogram if desired, due to the visual impact and intuitive appeal of histograms.) However displayed, this result may in turn be used as an operand in further arithmetic operations.

The representational technique of using a set of intervals — non-overlapping in the case of PDFs, singleton in the case of an interval operand, and overlapping in the case of results of arithmetic operations — and a probability mass associated with each interval in the set, is the key to our algorithm. This representation we term an *intermediate distribution* and it mediates between PDFs, intervals, and bounding CDFs and is the underlying representation used by the algorithm for its computations. A fuller description of the algorithm is given in Berleant (1993 [1]).

Related algorithms appeared beginning in 1968 (Ingram et al. [6]), however these algorithms were not automatically verifying. A recent constraint satisfaction approach to the problem is in Hyvönen (1995 [4]). An approach to using CDF bounds in computer systems administration appears in Post and Diltz (1986 [8]) under the rubric of the *stochastic dominance* field, which recognizes the usefulness of bounded families of CDFs. Another area of related work is robust statistics. We have not been able to find work by others discussing automatically verified operations where one operand is an interval and the other is a distribution function. A fuller review of related work appears in Berleant (1993 [1]).

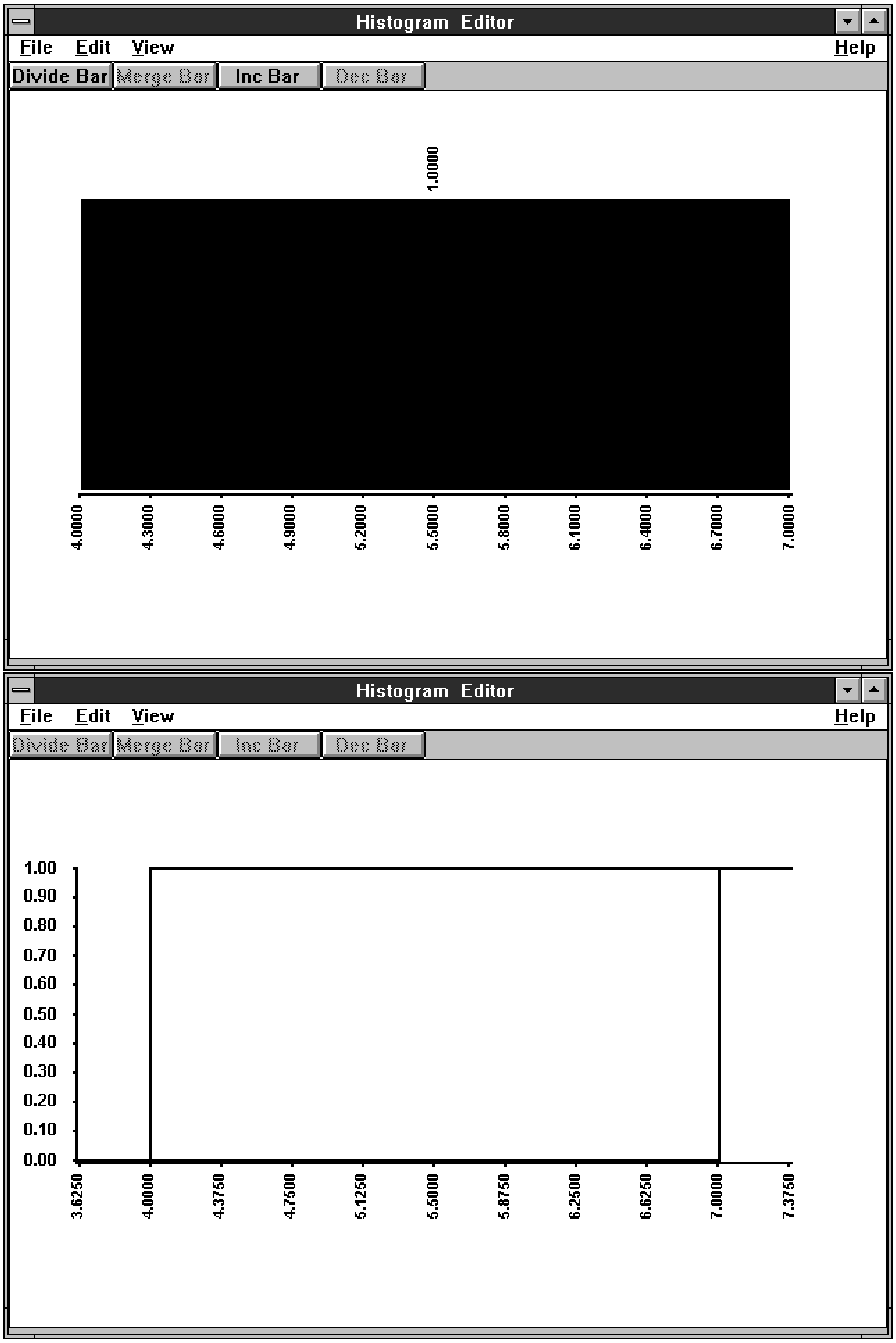

Figure 2: The interval [4, 7] and its representation as two CDFs bounding the family of all CDFs consistent with the interval.

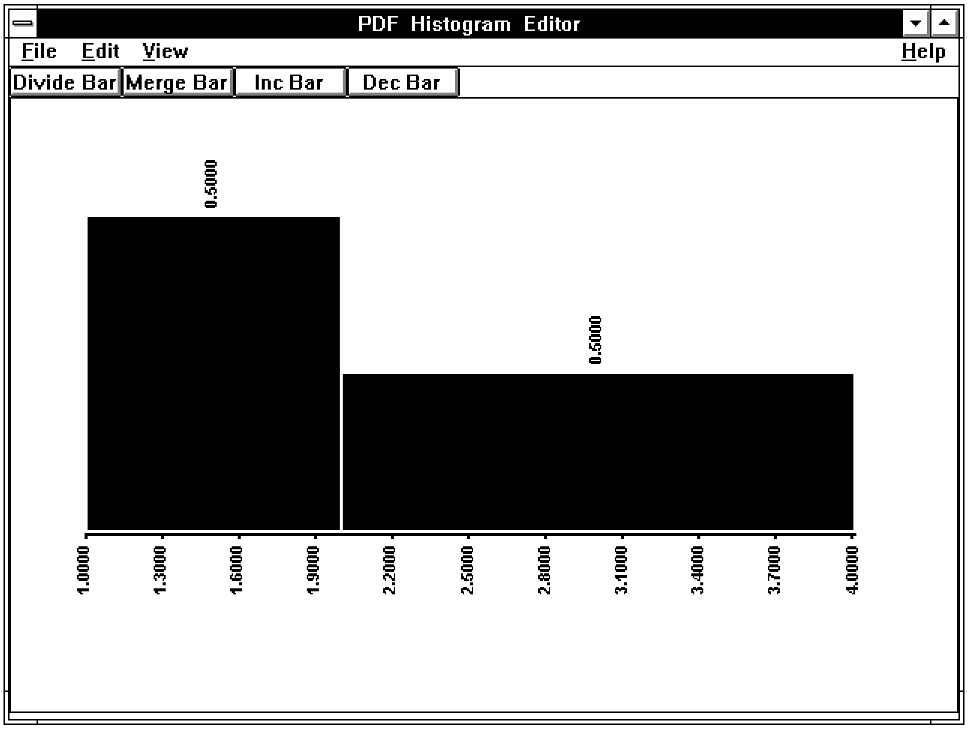

Figure 3a: Let this histogram describe uncertain value X. It depicts the intermediate distribution $\{p([1,2]) = \frac{1}{2}, p([2,4]) = \frac{1}{2}\}$.

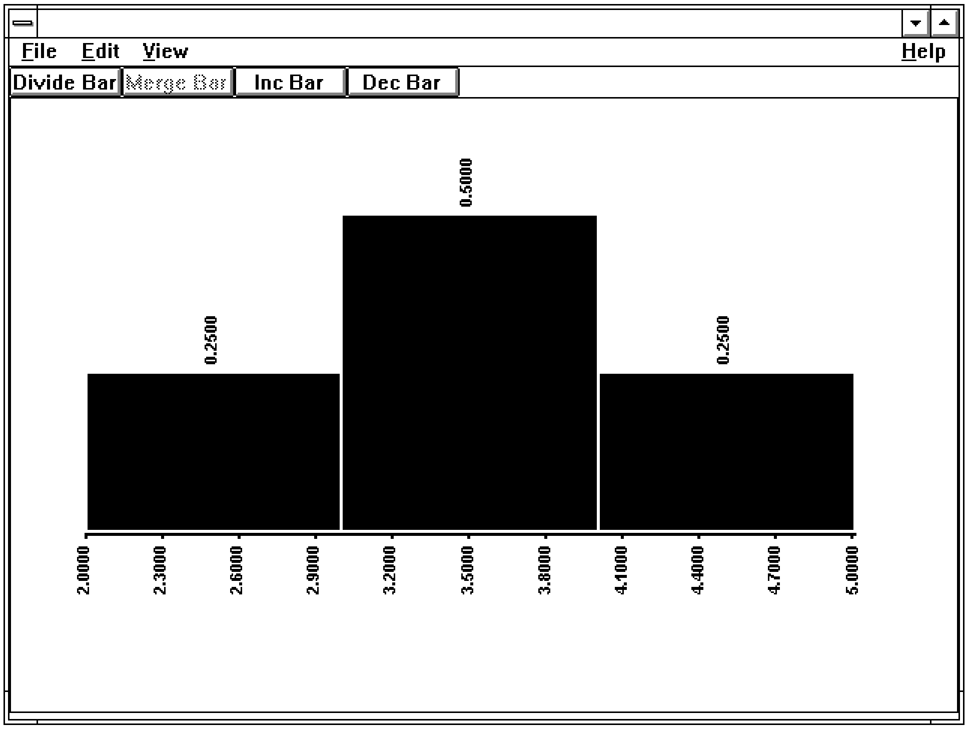

Figure 3b: Let this histogram describe uncertain value Y. It depicts the intermediate distribution $\{p([2,3]) = \frac{1}{4}, p([3,4]) = \frac{1}{2}, p([4,5]) = \frac{1}{4}\}$.

Cartesian product term #		Operand$_1$ (X_i)	Operand$_2$ (Y_j)	Result intermediate distribution
#1	interval probability	$[1,2]$ 1/2	$[2,3]$ 1/4	$[2,6]$ 1/8
#2	interval probability	$[1,2]$ 1/2	$[3,4]$ 1/2	$[3,8]$ 1/4
#3	interval probability	$[1,2]$ 1/2	$[4,5]$ 1/4	$[4,10]$ 1/8
#4	interval probability	$[2,4]$ 1/2	$[2,3]$ 1/4	$[4,12]$ 1/8
#5	interval probability	$[2,4]$ 1/2	$[3,4]$ 1/2	$[6,16]$ 1/4
#6	interval probability	$[2,4]$ 1/2	$[4,5]$ 1/4	$[8,20]$ 1/8

Figure 3c: Multiplying the intermediate distributions for X and Y leads to another intermediate distribution, shown in the last column. (This intermediate distribution contains overlapping intervals, as is typical of intermediate distributions resulting from arithmetic operations.)

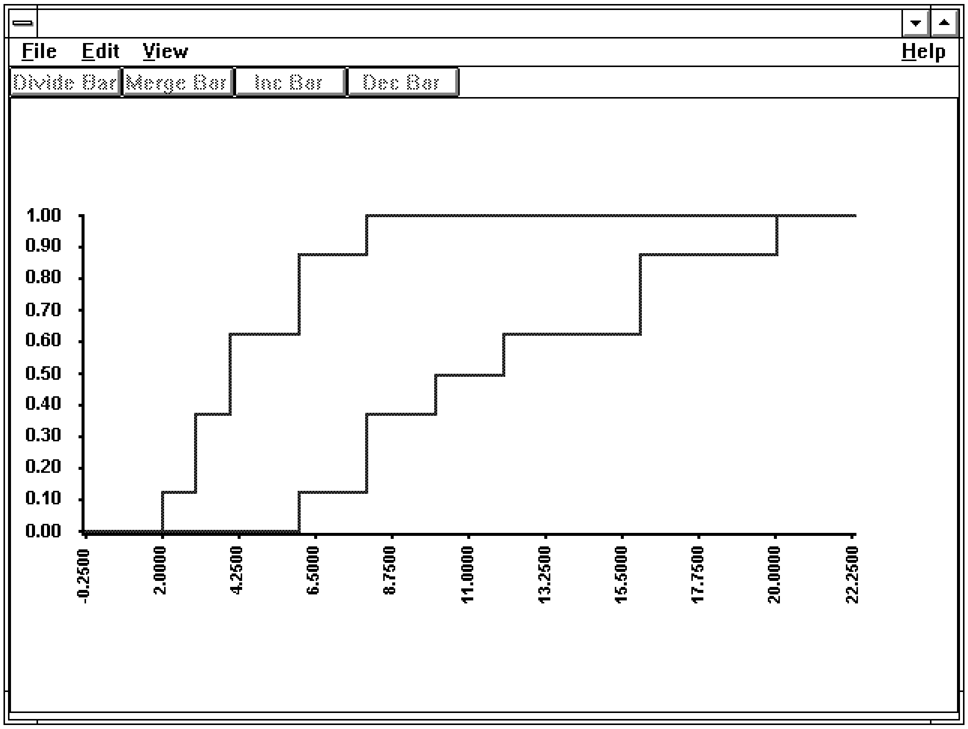

Figure 3d: Integrating the result intermediate distribution in the last column of Figure 3c produces these stair-step shaped bounding CDFs.

Figure 3 (a-d): Multiplying two operands. The user created two histograms X (Figure 3a) and Y (Figure 3b) using a graphical editor. These histograms are represented internally as intermediate distributions, that is, lists of intervals and associated probability masses. The intermediate distributions are used to calculate $X \times Y$, whose intermediate distribution is shown in the last column of Figure 3c. Integrating this result produces the CDF bounds shown in Figure 3d.

---

3 A Software Tool

A software calculating tool that runs under Microsoft Windows has been written. This tool does automatically verified arithmetic operations on operands, each of which may be either an interval or a distribution function.

3.1 Creating and Displaying Operands

The tool allows the user to graphically create and edit histograms using the mouse. A mouse-click based menu can be used to increase or decrease the number of bars in the histogram. Clicking the left button above a bar increases its height (and hence its area, which defines its probability mass) in proportion to how high above the bar the cursor is when the left button is pressed. Clicking the left button *below* the top of the bar decreases its height analogously. The width of a bar can also be increased or decreased, by clicking the right (instead of the left) mouse button above or below the top of the bar. As an alternative to mouse-based graphical editing, numerical values can be typed in directly when desired, by clicking on the Edit button at the top of the screen (visible e.g. in Figure 1) and invoking a dialog box. Clicking on the File button allows saving a histogram to a disk file, or reading one in from a disk file. The intent is to provide a user interface that is easy-to-use yet quite flexible.

An input histogram, as well as the result of an arithmetic operation on histogram operands, is represented internally as an intermediate distribution. This set of intervals each associated with a probability mass, when stored on disk, is stored as a list of triples. Each triple specifies specifying an interval low bound, an interval high bound, and a probability mass for the interval. This list can be edited using an ordinary text editor if desired. All intermediate distributions may be displayed graphically as histograms. These histograms are labeled Approximate when the intermediate distribution has overlapping intervals, as is usually the case for the result of an arithmetic operation. Approximate histograms cannot be edited, and like other histograms can alternatively be displayed correctly (not approximately) as a pair of bounding CDFs.

3.2 Operations on Operands

The tool provides several primitive operations. A menu of these operations is invoked by clicking on the Operation button at the top of the screen (Figure 4). Operations in $\{+, -, \times, \div\}$ take the top two panels A and B as operands and place the result in the lowest of the three panels, panel C (Figure 5), overwriting anything that may already be in C. If this result is to be used as an operand in a subsequent operation, it may be moved into panel A or B using one of the Exchange operations (Figure 4). An operand or result may be viewed as either a histogram or a pair of CDF bounds and may be viewed in detailed form (e.g. Figure 1), with or without a grid.

As an added feature, the tool can apply results of Kolmogorov (1941 [7]) and others to obtain bounds on the CDF for a random variable for which some samples are provided. The samples must be provided in a text file ending in "`.smp`" and containing their numerical values. This file may be read into panel A and displayed in cumulated form as an empirically estimated CDF for the underlying random variable. Then, the operation C = A's Confidence Limits ...

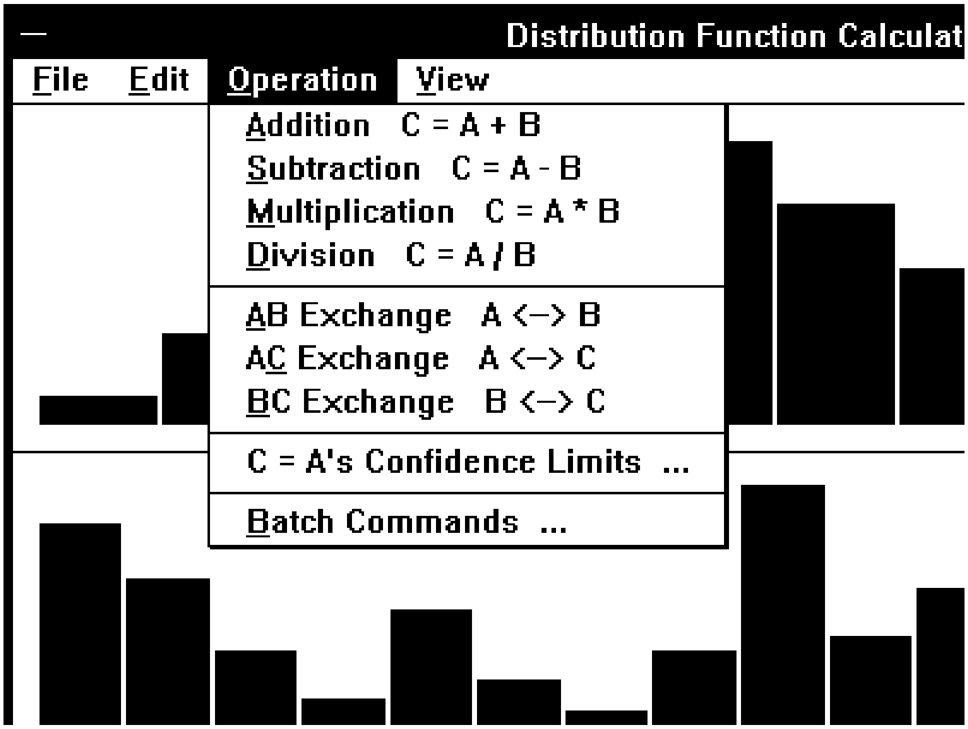

Figure 4: The operations offered by the tool.

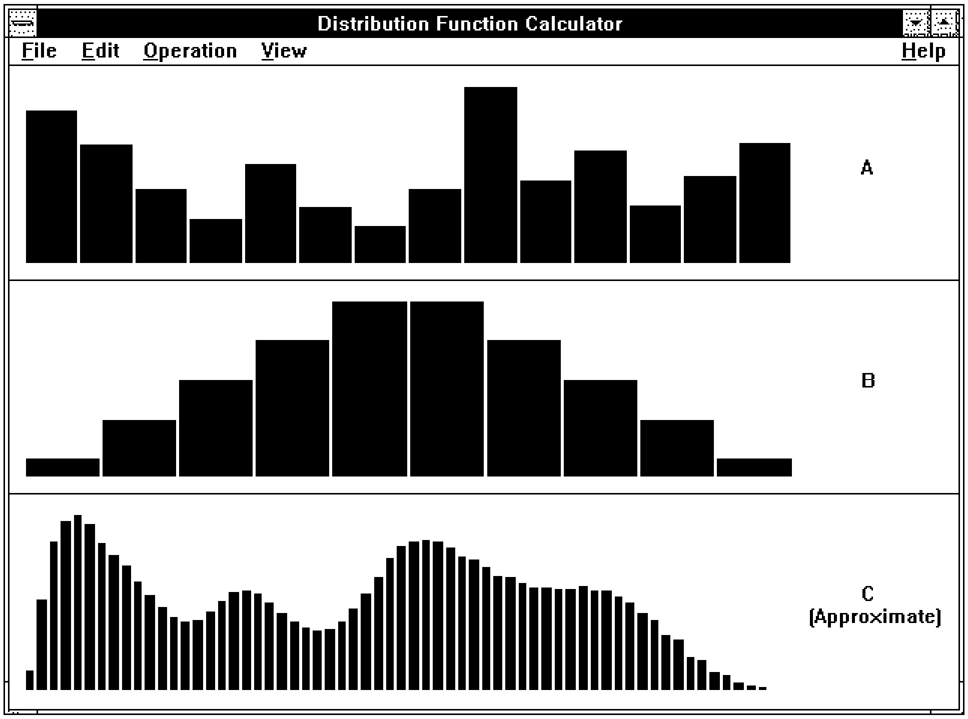

Figure 5: Panel C contains the result of applying an operation (division in this case) to A and B. Panel C, labeled Approximate as displayed in histogram form, can alternatively be exactly displayed as a pair of bounding CDFs.

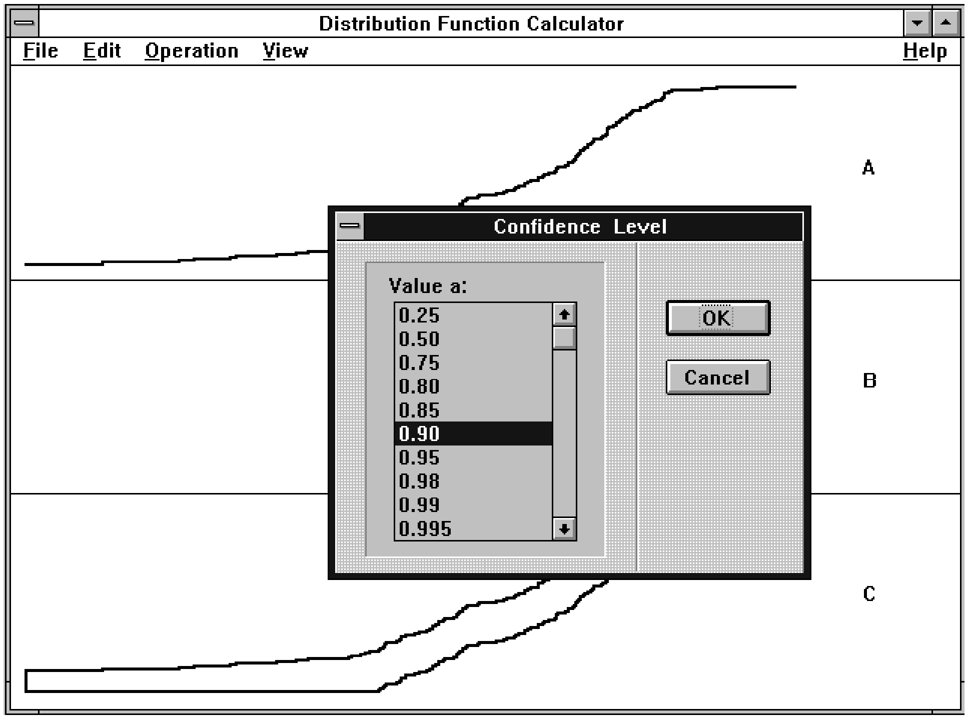

Figure 6: Confidence limits around the CDF in panel A, if generated from samples of a random variable (several dozen student test scores in this case) may be calculated and shown in panel C. The confidence level is chosen from a menu.

(Figure 4) may be invoked to place in panel C bounds on the actual but unknown CDF for the random variable, to a given confidence level (Figure 6).

Various functions of the software tool may be grouped together and partially automated using a primitive batch command facility.

4 Future Work

We now have a tool for doing common arithmetic operations on distribution function and interval operands. However, much more remains to be done to extend and apply the work.

A next major stage in the research is to identify applications and apply the tool to those applications. Once identified, an application can serve not only to demonstrate the practical value of an idea but also to guide its further extension. Digital signal processing and highway maintenance decision support are among those that have been suggested. Applications of stochastic dominance are also possible candidates, as are applications of robust statistics. The world is full of incompletely known values which need to be taken into account, underlining the importance and potential value of research dealing with incompletely known information.

A number of extensions to the tool described here would be useful:

- Handling calculations whose intermediate results are dependent on one another could be done by implementing not only the common arithmetic operations as we have done, but also combinations of those so that, for example, given histograms in panels A and B, $(B-A)/A$ could be calculated even though $(B-A)$ and A are dependent on each other. This would require a simple expression parser and, although excess width might occur in constituent interval calculations in many cases, the result would still be correct (Berleant 1993 [1]).

- Providing exponentiation, logarithms, trigonometric and other unary and binary operations would be useful.

- Handling distribution functions that have tails extending out to infinite could be handled by applying the work of others on handling intervals containing bounds of ∞ and $-\infty$.

- The tool is presently modeled after a calculator, in which only a small number of operands can be handled at once. A useful extension to the calculator concept is the spreadsheet concept. Interval mathematics has been implemented in spreadsheets (Hyvönen 1994 [5]) and distribution functions have also been shown to be feasible as cells in spreadsheets in commercial products such as Crystal Ball [3] and @RISK [9].

5 Conclusion

The present paper describes a computer tool with graphical user interface capabilities, for allowing users to specify interval and distribution function operands as histograms and allowing them to perform automatically verified arithmetic operations on those operands. A previous paper (Berleant 1993 [1]) reviews related work and describes in more detail how automatically verified arithmetic operations may be carried out on intervals and distribution functions. An important next stage in this research is to find applications for the techniques in order to demonstrate practical use of the tool and its underlying ideas.

Details on the implementation appear in Cheng (1994 [2]). The software is available at no charge (for non-commercial purposes) from the authors.

6 Acknowledgements

We thank Robert Babcock for helping to facilite support for one of the authors (Cheng), Vladik Kreinovich for his encouragement, Dan Layne for originally suggesting the interval mathematics forums, and Bill Walster for suggesting some possible application directions.

References

[1] D. Berleant, "Automatically Verified Reasoning with Both Intervals and Probability Density Functions," *Interval Computations,* 1993 No. 2, pp. 48–70.

[2] H. Cheng, "A Software Tool for Automatically Verified Reasoning with Intervals and Cumulative Distribution Functions," Master's Thesis, Dept. of Computer Systems Engineering, U. of Arkansas, Fayetteville, Dec. 1994.

[3] Crystal Ball, software product, Decisioneering Inc., 1380 Lawrence St. Suite 520, Denver CO 80204.

[4] E. Hyvönen, "Constraint Reasoning on Probability Distributions," manuscript (1995), author is at VTT Information Technology, P.O. Box 1201, 02044 VTT, FINLAND, `eero.hyvonen@vtt.fi` .

[5] E. Hyvönen, "Spreadsheets Based on Interval Constraint Satisfaction," *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing,* Vol. 8, 1994, pp. 27–34.

[6] G. E. Ingram, E. L. Welker, and C. R. Herrmann, "Designing for Reliability Based on Probabilistic Modeling Using Remote Access Computer Systems," *Proceedings 7th Reliability and Maintainability Conference,* American Society of Mechanical Engineers, 1968, pp. 492–500.

[7] A. Kolmogoroff (*a.k.a.* Kolmogorov), "Confidence Limits for an Unknown Distribution Function," *Annals of Mathematical Statistics,* Vol. 12, No. 4, 1941, pp. 461-463.

[8] G. V. Post and J. D. Diltz, "A Stochastic Dominance Approach to Risk Analysis of Computer Systems," *Management Information Systems Quarterly,* Vol. 10, No. 4, Dec. 1986, pp. 363–375.

[9] @RISK, software product, Palisade Corp., 31 Decker Rd., Newfield, NY 14867.

Contact: Daniel Berleant
Dept. of Computer Systems Engineering
University of Arkansas
Fayetteville, AR 72701
Email: djb@engr.uark.edu