Grid-based Histogram Arithmetic for the Probabilistic Analysis of Functions*

Carlos Carreras+ and Manuel V. Hermenegildo*

+ *Department of Electrical Engineering*
* *Department of Computer Science*
Technical University of Madrid
28040 Madrid, SPAIN
carreras@die.upm.es, herme@fi.upm.es

* This work was funded in part by project CICYT TIC97-0928

Overview

- Motivation
- Intervals from an abstract interpretation perspective
- Interval grids as abstractions
- Interval histogram grids as abstractions
- Grid-based histogram arithmetic
- Comparison of approaches through an illustrative example
- Conclusions and future work

Motivation

Objectives

- Probabilistic characterization of functions, range computation, optimization
- Applications: electronic design, reliability analysis, constraint solving, robotics

Approaches

- Theoretical methods: exact (not general)
- Exhaustive simulation: all possible inputs (small input spaces)
- Partial simulation: random/test vectors (worst-case not guaranteed, slow)
- Range arithmetic: fast (large errors, no value dependencies)
- **Grid-based interval (histogram) simulation**

Basic Definitions

Interval $[a, b]$**:** Set of $N = b - a + 1$ integers x that verify $a \le x \le b$.

Interval operations: Based on computations on the endpoints (> 0):

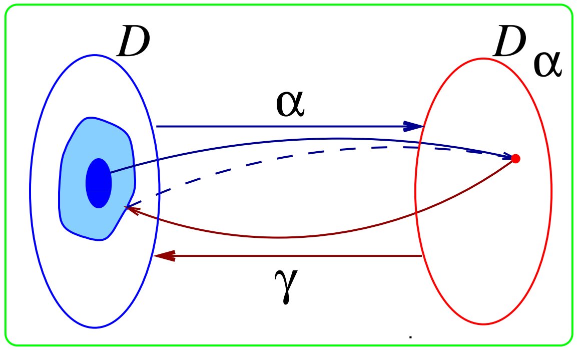
$$
\begin{aligned}
[x_1, y_1] + [x_2, y_2] &= [x_1 + x_2, y_1 + y_2] \\
[x_1, y_1] - [x_2, y_2] &= [x_1 - y_2, y_1 - x_2] \\
[x_1, y_1] \times [x_2, y_2] &= [x_1 \times x_2, y_1 \times y_2] \\
[x_1, y_1] \div [x_2, y_2] &= [x_1 \div y_2, x_2 \div y_1]
\end{aligned}
$$

Operations on sets of intervals: Computation of the Cartesian product and merging of overlapping results in a single interval

Abstract Interpretation Using (Sets of) Intervals

Basic concepts:

- Concrete domain $(D, +, \times, ...)$: integers
- Abstract domain $(D_\alpha, +_\alpha, \times_\alpha, ...)$: intervals
- Abstraction function: $\alpha : D \mapsto D_\alpha$
- Concretization function: $\gamma : D_\alpha \mapsto D$
- Partial order: $\langle D, \subseteq \rangle$ and $\langle D_\alpha, \sqsubseteq \rangle$



Properties: Impact of data dependencies

$$a) \quad \text{Safety}: \quad a + b \subseteq \gamma(\alpha(a) +_\alpha \alpha(b))$$
$$b) \quad \text{Partitioning}(\alpha'): \quad \alpha(a) +_{\alpha'} \alpha(b) \sqsubseteq \alpha(a) +_\alpha \alpha(b)$$

Interval Grids as Abstractions

Interval grid

- Partition of D in terms of intervals I_i : $G = \{I_i | \cup_{\forall i} I_i = D, \cap_{\forall i} I_i = \emptyset\}$
- Types: custom and analytic (linear, geometric)
- Analytic grids are characterized by a granularity g
- Set of intervals allowed by a grid G, $I_G = \{i | i \in I, i \leq j, j \in G\}$

Abstraction

- G induces an abstract domain $D_{\alpha,G}$ defined by 2^{I_G}
- α_G maps all concrete values in the same I_i to a single element of $D_{\alpha,G}$

Linear Interval Grids

- Intervals of equal size g (except around the origin) identified by level i
- More appropriate for additions, subtractions and multiplications by constant
- Complex models: adjustable center and focus

$$I_i = \begin{cases} [gi+1, g(i+1)] & (i < -1) \\ [-g+1, -1] & (i = -1) \\ [0, 0] & (i = 0) \\ [1, g-1] & (i = 1) \\ [g(i-1), gi-1] & (i > 1) \end{cases}$$

Example: $G_{lin(5)} = \{\ldots, [-4, -1], [0, 0], [1, 4], [5, 9], [10, 14], \ldots\}$

$$\{3, 4, 6, 9, 11\} \subset D \Rightarrow \{[3, 4], [6, 9], [11, 11]\} \subset D_{\alpha,G}$$

Geometric Interval Grids

- Interval size increases exponentially (powers of g) away from the origin
- Intervals are identified by level i
- More appropriate for multiplications, divisions and exponentiations
- Complex models: adjustable center and focus

$$I_i = \begin{cases} \left[-g^{-i}+1, -g^{-(i+1)}\right] & (i < 0) \\ [0,0] & (i = 0) \\ \left[g^{(i-1)}, g^i - 1\right] & (i > 0) \end{cases}$$

Example: $G_{geo(2)} = \{\ldots, [-1,-1], [0,0], [1,1], [2,3], [4,7], [8,15], \ldots\}$

$$\{3,4,6,9,11\} \subset D \Rightarrow \{[3,3],[4,6],[9,11]\} \subset D_{\alpha,G}$$

Histogram Grids as Abstractions

Generalized interval $[a, b]/p$**:** Set of $N = b - a + 1$ integers x that verify $a \leq x \leq b$ with an associated probability mass p (uniformly distributed)

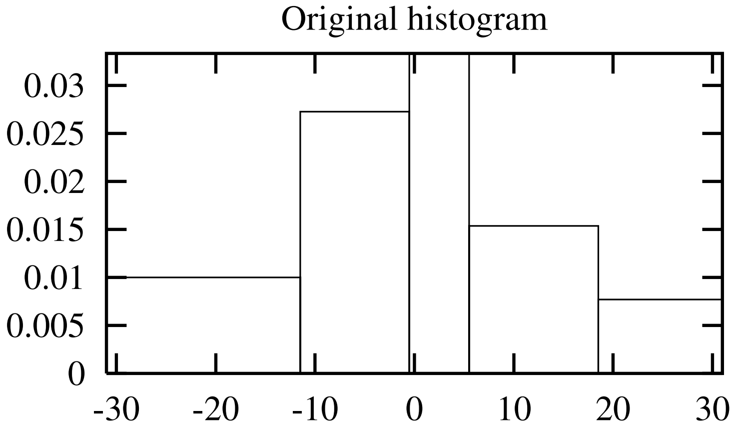
Interval-based histograms

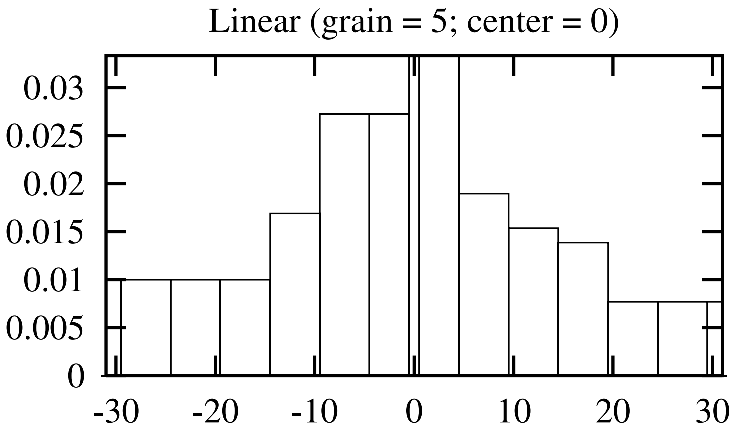
- Input space size dependent on the available data (input histograms)
- Output collection times become too large (two intersecting results produce three output intervals with different probabilities)

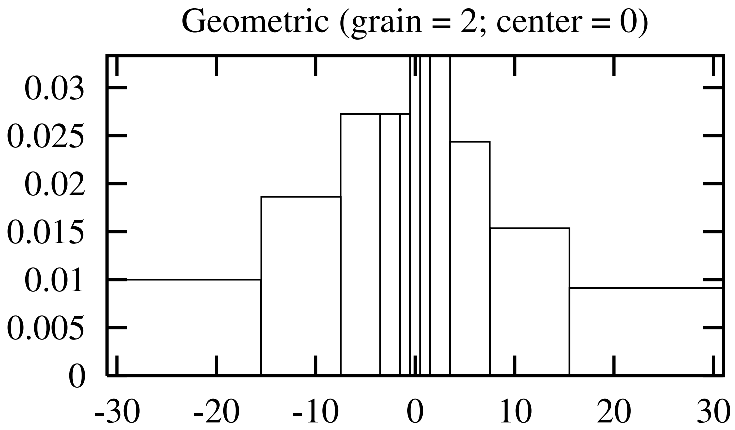
Grid-based histograms

- *Split rule:* Intervals crossing I_i bounds are split
- *Merge rule:* Intervals inside the same I_i are merged

Examples of Grid-based Histograms







Computation Model based on Interval Arithmetic

Algorithm

1. Input space: $N_1 \times \cdots \times N_I$ where $N_i =$ set of intervals of G_i-histogram i
2. For each input vector $(\ldots, [a_{ij}, b_{ij}]/p_{ij}, \ldots)$:
   (a) Compute its probability $P = \prod_{i=1}^{I} p_{ij}$
   (b) Operate using interval arithmetic
   (c) Assign P to each resulting interval
   (d) Collect the results in output G_o-histograms (split and merge rules)

Advantages: Control of input space size, fast collection of results

Problem: Accuracy of probabilities after some operations (mult, exp)

Grid-based Histogram Arithmetic

Arithmetic: Alternative (approximate) operators that provide $G-$based results

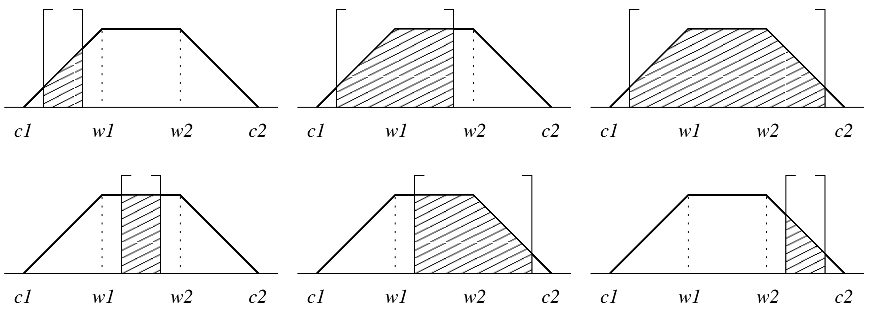
Modified computation model: (operation result = several bars)

1. Input space: $N_1 \times \cdots \times N_I$ where $N_i =$ set of intervals of G_i-histogram i
2. For each input vector $(\ldots, [a_{ij}, b_{ij}]/p_{ij}, \ldots)$:
   (a) Compute its probability $P = \prod_{i=1}^{I} p_{ij}$
   (b) Transform each $[a_{ij}, b_{ij}]/p_{ij}$ into $[a_{ij}, b_{ij}]/1$
   (c) For each operation and for each combination of its inputs:
      i. Operate using G-based histogram arithmetic (single-bar operands)
      ii. Proceed with next operation, if any (step c). Then return.
      iii. Multiply by P the probability of each resulting interval
      iv. Collect the histogram results in output G_o-histograms

Addition/Subtraction Model (G = geometric, 2)

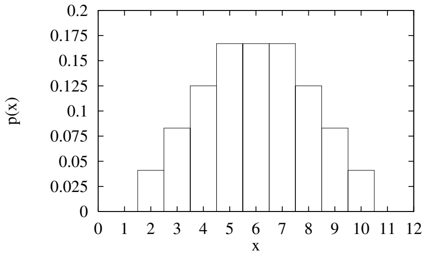
Theorem:

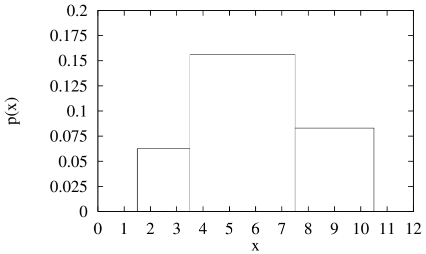
$$[]_{l_a>0} + []_{l_b>0} \subset []_{max\{l_a,l_b\}}, []_{max\{l_a,l_b\}+1}$$



Exact Model:

- Subtraction: $-[a,b]_l = [-b,-a]_{-l}$
- Positions of the *pdf*$_+$ vs G
- Distribution of the $N_a N_b$ results



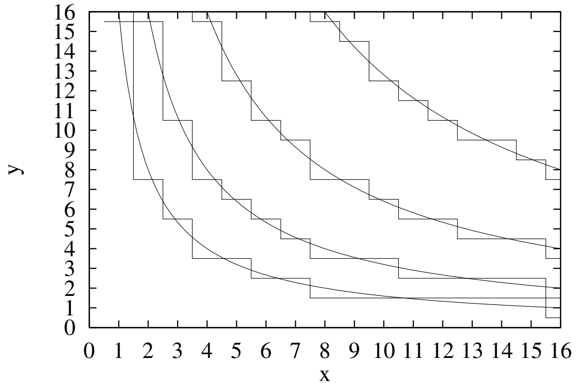


Example: $[-7,-4]/1 + [9,14]/1$

Multiplication/Division Model (G = geometric, 2)

Theorems:

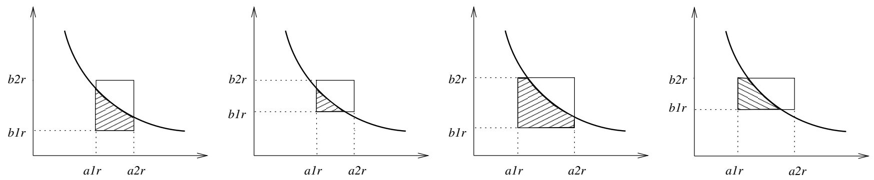
$$
\begin{aligned}
[]_{l_a} \times []_{l_b} &\subset []_{l_a+l_b-1}, []_{l_a+l_b} \\
[]_{l_a} \div []_{l_b} &\subset []_{l_a-l_b}, []_{l_a-l_b+1}
\end{aligned}
$$

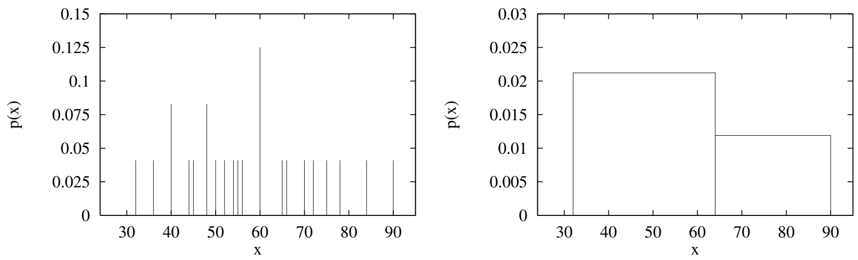


Approximate Model:

- Independent sign computation
- Area ($\Re$) below the curves:

$$
\begin{aligned}
x \times y &= 2^{l_a+l_b-1} \\
x \div y &= 2^{l_a-l_b}
\end{aligned}
$$





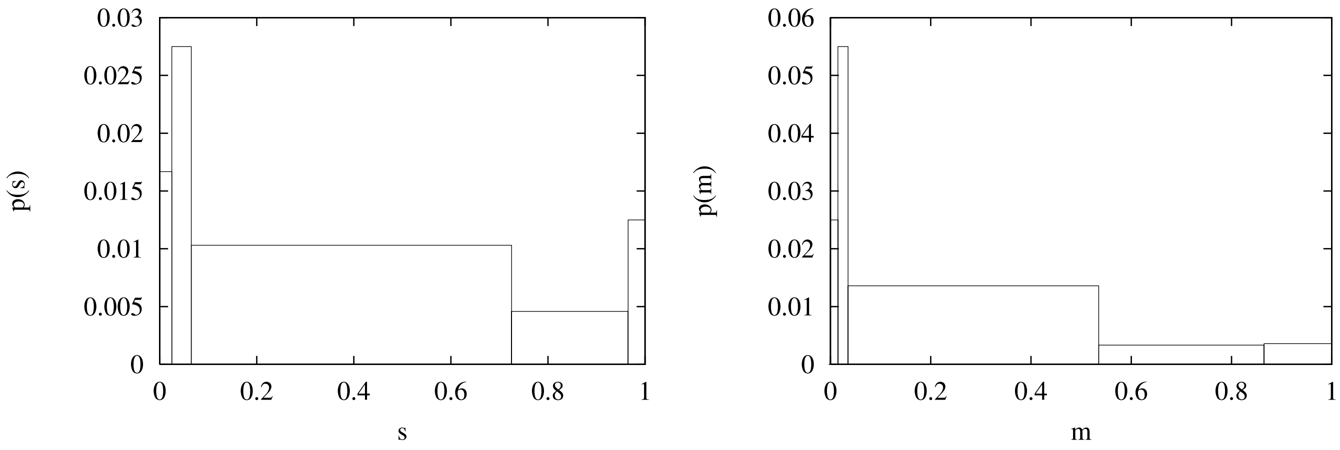
Example: $[4, 6]/1 + [8, 15]/1$

Example of a Computation

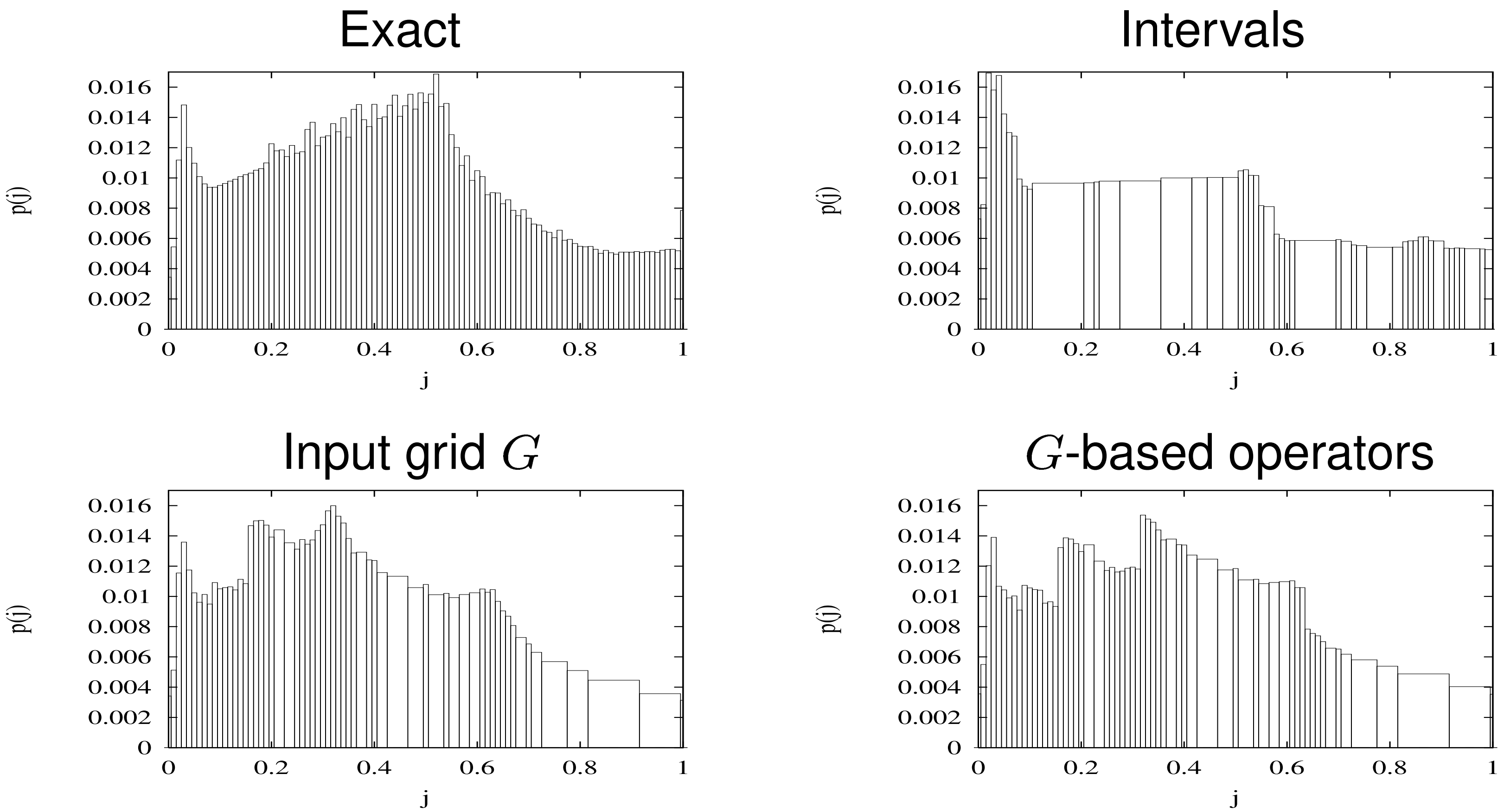
- Reliability estimation of a robot arm joint:

$$j = s_a \times s_b + m - s_a \times s_b \times m$$

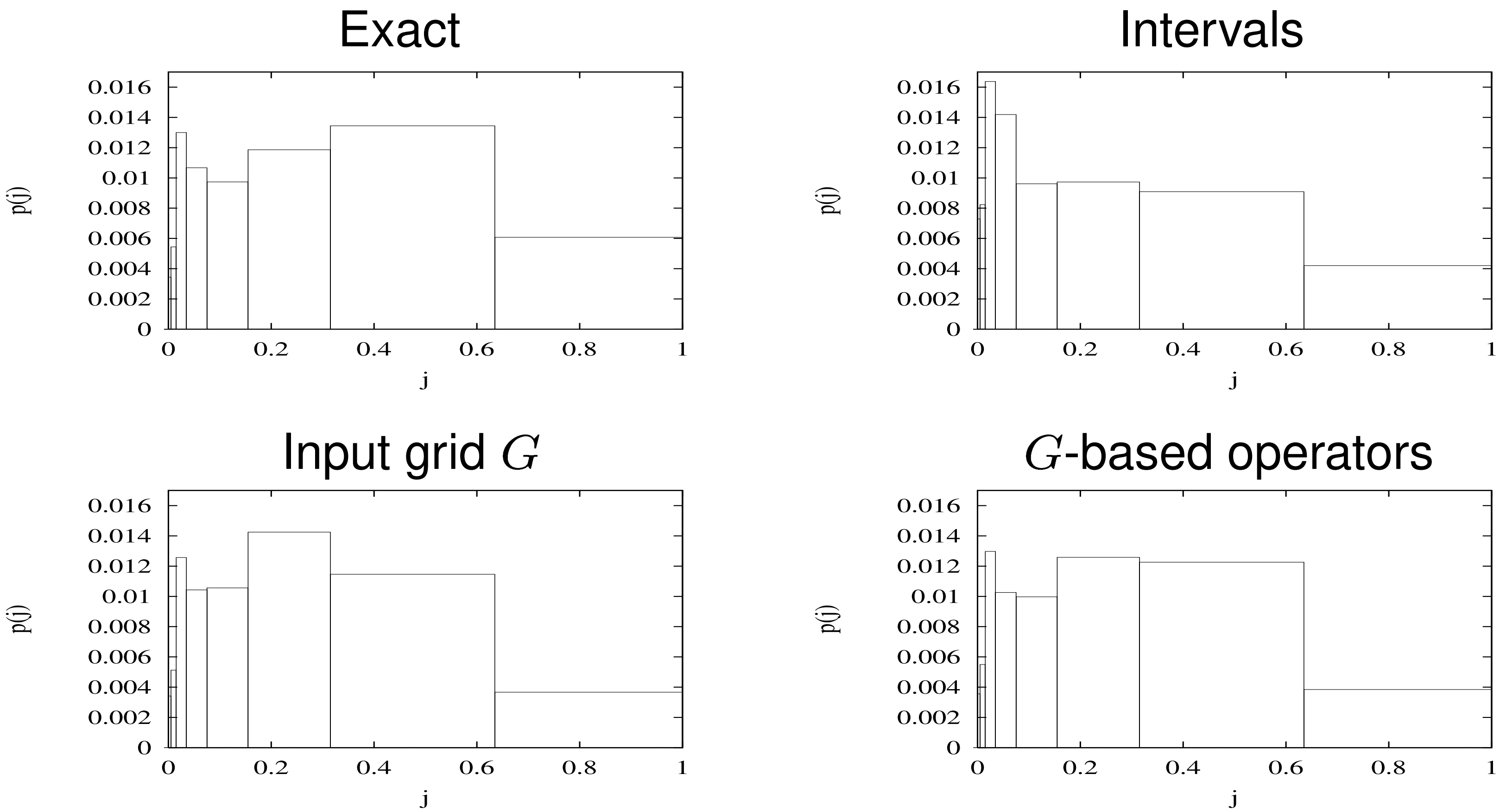
- Probabilities of failure: j (joint), s_a, s_b (sensors), and m (motor)
- Scaling by 10^2 (exhaustive exploration: 10^6 input vectors of integers)



Collection Strategy: Merging of Individual Results



Collection Strategy: Output Grid G



Statistics from the Computations

Error definition: (for G-based representations)

$$\mathsf{Error}(\%) = 100 \times \sum_{\forall \mathsf{bars}} \frac{|\ \mathsf{Exact} - \mathsf{Approx}\ |}{2 \times \mathsf{Exact}}$$

Input	Computation	Output	Size	Time	Memory	Error (%)
integer	integer	no grid	6181806	2773459	20844	-
integer	integer	geometric(2)	6181806	2297749	10651	0
no grid	no grid	no grid	750	490	739	-
no grid	no grid	geometric(2)	750	240	602	17.42
geometric(2)	no grid	no grid	3072	1640	1159	-
geometric(2)	no grid	geometric (2)	3072	820	642	6.55
geometric(2)	geometric(2)	no grid	3072	2120	1490	-
geometric(2)	geometric(2)	geometric(2)	3072	1260	642	3.94

Conclusions and Future Work

- Grid-based histogram method for probabilistic characterization
- Controlable tradeoff between accuracy and computation times
- Support for exhaustive exploration of input space (vs partial)
- Grid-based operators for increased probability accuracy
- Better solution than using larger input spaces
- Selective use of these operators can be very effective

- More testing combining standard and grid-based operators
- Identifying critical points to apply grid-based operators
- Custom grid generation to improve effectiveness