

Finding the Likely Behaviors of Static Continuous Nonlinear Systems

Alexander Yeh

MIT Lab. for Computer Science, 545 Technology Sq., NE43-413, Cambridge, MA
02139, USA

phone: (617) 253-3447, electronic mail: ay@zermatt.lcs.mit.edu

March 12, 2008

keywords & phrases: Monte Carlo, probability bound, uncertainty propagation,
uncertainty representation

Abstract

This paper describes two methods for predicting the likely behaviors of static continuous nonlinear systems with varying input values. The methods use a parameterized equation model and upper or lower bounds on the joint input density to bound the likelihood of a behavior, such as a state variable being inside a numeric range. Using a bound on the density instead of the density itself is desirable because the density's parameters and shape are not exactly known. The first method is limited to using lower density bounds. It finds rough bounds at first, and then refines them as more iterations of the method are allowed. The second method is a hit-or-miss version of sample-mean Monte Carlo. Unlike the first method, the second method can also handle upper density bounds, which are more useful than lower density bounds, but the generated probability bounds are only approximate. However, standard deviations on the bounds are given and become small as the sample size increases. In contrast to other researchers' methods, the two methods described here (1) find all the possible system behaviors, and tell how likely they

are, (2) do not just approximate the distribution of possible outcomes without some measure of the error magnitude, (3) do not use discretized variable values, which limit the events one can find probability bounds for, (4) can handle density bounds, and (5) can handle such criteria as two state variables both being inside a numeric range.

1 Introduction

This paper describes two methods to predict the likely steady-state behaviors of a continuous nonlinear system in which the input values can vary. The methods use a parameterized steady-state equation model and upper or lower bounds on the joint input probability density to bound the likelihood that one or more state variables stay inside or outside a given set of numeric ranges. The first method is called SAB, which stands for *split and bound* (the method's basic steps). SAB utilizes lower density bounds only and produces analytical likelihood bounds. The second is a form of Monte Carlo and is called HMC, which stands for *hit-or-miss sample-mean Monte Carlo*. HMC can utilize both lower and upper density bounds and produces estimates of likelihood bounds with standard deviations on those estimates.

The reason for using one or more bounds on the probability density, and not the density itself, is that density parameters (means, etc.) and density shape are often not exactly known: One may only have confidence intervals or estimated ranges for some parameters. In fact, the actual parameters may vary (cyclically) over time, such as the mean blood pressures in the chest which rise and fall due to pressure changes caused by the breathing cycle. Even if one has point estimates for all the parameters, they may be unusable due to slight inconsistencies. For

example, the correlation estimates may be slightly inconsistent, especially if some correlations are estimated using intuition or heuristics, as opposed to applying statistics over a large data-base. By relaxing correlation estimates to be ranges of values around the original estimates, one can examine densities with that type of correlation structure. For example, if some correlation coefficient estimate of 0.8 was found to be inconsistent with the other density parameters, one could still do preliminary analysis with a density where that correlation coefficient is high (near 0.8) by allowing that coefficient to be anywhere in the range of 0.7 to 0.9 (assuming some value within this range is consistent with all the other density parameters).

An alternative to using bounds on the probability density is to use a sample of possible probability densities. However, sampling is not as complete an examination as bounding. Some class of important behaviors may lie between all the samples and not be observed.

Other prediction-making methods have one or more of the following problems: not finding the likelihood of behaviors or only finding the likelihood of the variable values falling in certain ranges; not estimating the result's error and not being able to improve on an initial result's accuracy when given more computation time; not being able to handle density bounds, or handling them too slowly; needing all the set(s) of input values that satisfy the criteria to be explicitly mentioned.

Compared to these other techniques, the two methods described in this paper can produce estimates of their errors, improve their answers as more samples or iterations are allowed, and deal with distributions of continuous variable values. SAB produces analytical answers but can only handle lower density bounds. HMC is a variation on the Monte Carlo techniques that integrate the density bound. The variation makes it possible to determine the interval to be integrated over.

The next section of this paper gives a more detailed description of some other methods. Section 3 gives a simple example of using the two methods described in this paper. It is followed by three sections which give in order an overview of SAB, a demonstration of how SAB runs in the simple example, and some of SAB's problems.¹ After this is a section on HMC, which describes how it is derived from and differs from sample-mean Monte Carlo, and then a section on numeric bounding techniques needed for both SAB and HMC. The paper ends with a discussion section.

2 Other Techniques

Other prediction-making methods fit into one of four categories. The first category of methods finds all the possible system behaviors (sometimes including impossible ones), but does not tell the likelihood of the behaviors. Such methods include systems either performing qualitative reasoning [1, 15], or providing numeric bounds [12].

Category two methods estimate the distributions of possible outcomes without giving some measure of each estimate's error and will not improve the accuracy of those estimates when given more computation time. One of these methods is using moments [3]. This method uses truncated Taylor series expansions of the model equations to find various moments (mean, variance, etc.) of the distributions of interest.

A third category is evidential reasoners [6, 14], which include most of the current work done on uncertainty in AI. These reasoners can only handle a variable

¹SAB's details can be found in [16].

value in terms of the possibility of it belonging to one or more regions in a preset discretization of the possible variable values. For example, blood pressure (*bp*) may be only thought of in terms of being low, normal, or high. This limitation is a problem because what is considered normal, desirable, etc. can change with each use of a model. For example, when trying to lower a patient's *bp*, an acceptable pressure depends on the patient's former normal blood pressure and the patient's ability to withstand therapy side-effects.

Monte Carlo techniques [3, 5, 7], which fall into two general classes, constitute the fourth category. The first class simulates a system by generating samples according to some probability distribution. Most methods in this class cannot handle density bounds. The acceptance/rejection method can handle density bounds, but it is too slow due to the large number of potential samples it rejects. The second class of Monte Carlo techniques integrates the density or density bound involved. These integration techniques include hit-or-miss and sample-mean Monte Carlo. Unfortunately, determining the interval(s) to be integrated over (the region(s) satisfying the criteria) is very hard. The section on HMC gives more details on this problem. Also, as with all Monte Carlo techniques, every answer is inexact and has a standard deviation associated with it.

3 Simple Example Using *PVR*

A simple example of using the two methods described in this paper involves finding a patient's pulmonary vascular resistance (*pvr*) given the constraint

$$pvr = (pap - lap)/co \tag{1}$$

and information on the patient's pulmonary arterial pressure (pap), left atrial pressure (lap) and cardiac output (co). pvr is of interest because a high value indicates that the heart's right ventricle has to work very hard to keep the blood moving through the lungs [10, p. 234]. One threshold condition is $pvr \leq 1.62mmHg/(l/min)$. Critically ill surgical patients with values above this are less likely to survive [13, p.54-59].² pap , lap , and co have patient and time dependent values, and are not easy to measure accurately. Table 1 gives statistics for the patient of interest, a heart attack victim. The question is, given information on pap , lap , and co for the patient involved, is pvr at all likely to be above the threshold? If so, one ought to monitor pvr .

The numbers are close enough so that the answer is not obvious from looking at Table 1: For example, substituting the mean values into Equation 1 results in $pvr < 1.62$, but increasing pap 's value in the substitution by 3.38 (one standard deviation) while maintaining lap and co 's values would result in $pvr > 1.62$. However, the latter is not that likely to happen because lap tends to increase when pap does (high positive correlation).

So, one has to look at the joint density of pap , lap , and co . Like most statistics, the ones in Table 1 are subject to sampling error, and in addition, the density shape is not exactly known. To get around this difficulty, one can hypothesize plausible bounds on the joint density and bound the probabilities of satisfying the criteria given each density bound. Ideally, the set of density bounds used will cover all the possible variations.

In this example, four density bounds are considered. They show the types of bounds the methods can handle. One-dimensional (marginal) views of these are in

²Assume the patient to have a body surface area of $1.74m^2$, the average for a human.

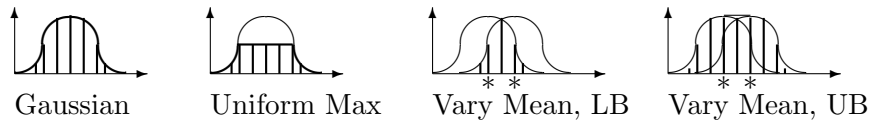


Figure 1: 4 Density Bounds

Figure 1, where the areas under the density bounds are marked by vertical lines. The one on the right is an upper bound, the middle two are lower bounds, and the one on the left can be either. The three left-most input bounds are given to SAB, and the right-most two are given to HMC. As will be described later, the two right-most bounds cover all Gaussian densities where co 's mean is somewhere within a bounded interval and all the other parameters are as given in Table 1. The details on producing the results are given later.

The first “bound” is a regular joint Gaussian density³ with the parameters listed in Table 1 and a one-dimensional view of it is shown in the left diagram of Figure 1. A 1000-sample Monte Carlo simulation with this bound (a normal probability density) indicates that $pvr > 1.62$ about 20% of the time. SAB analytically bounds this to be between 4% and 57%. This is consistent with the Monte Carlo simulation and with patient data, where 4 of 17 (23.5%) data points had $pvr > 1.62$.⁴

The second density bound is a

1. joint Gaussian density with the parameters listed in Table 1
2. in which the maximum density value is limited to that of a jointly uniform density with the same means and standard deviations.

³Being a density (has an area of 1), it is both a lower and an upper density bound.

⁴Here, the data could have been used by itself to answer the question of whether $pvr > 1.62$ is at all likely. SAB and HMC are meant to be used when such data are not available.

In other words, the density bound looks like a Gaussian far from the variables' means, but has the low flat top of a uniform density near the means. A one-dimensional view of it is shown in the middle-left diagram of Figure 1. Integrating the bound indicates that it includes $\sim 70\%$ of the probability mass. Using this bound SAB analytically bounds $\Pr(pvr > 1.62)$ to be between 4% and 79%. SAB found the 79% figure by finding a lower bound on $\Pr(pvr \leq 1.62)$ and then subtracting it from 100%. These results are again consistent with the patient data.

The third density bound is the lower bound of a Gaussian density where co 's mean is allowed to be anywhere between 6.20 to 6.78.⁵ This constraint might have been determined by using information in some confidence interval for co 's mean. The middle-right diagram of Figure 1 shows a one-dimensional view of this bound: co 's mean can lie anywhere between the two *'s. The lower density bound is the intersection of the areas under all the densities possible due to allowable variations in co 's mean. Because Gaussian densities are unimodal, the lower bound is the intersection of the areas under the two Gaussian density curves⁶ shown. Integrating the bound indicates that it includes $\sim 65\%$ of the probability mass. Using this bound SAB analytically bounds $\Pr(pvr > 1.62)$ to be between 1% and 76% and HMC estimates it to be between 12% and 44% with an estimated standard deviation of $< 2\%$ on both figures. Both these results are also consistent with the patient data.

The fourth density bound is the upper bound version of the third density bound. Instead of taking the intersection of the areas under the Gaussian density curves with co means in $[6.20, 6.78]$, take the union. HMC estimates that $\Pr(pvr > 1.62) <$

⁵The variances, covariances, and other means could also be allowed to vary.

⁶They are the ones with the extreme co mean values.

33% (standard deviation of 1%). As with all upper bounds, all of the probability mass is within the bound.

$\Pr(pvr > 1.62) > 1\%$ is consistent with all four input bound cases, so pvr should be monitored. Note that the SAB results can be tightened. See the end of Section 5 for details. Similarly, HMC’s estimates can be made more accurate.

4 SAB (Method 1): Overview

This section gives an overview of SAB’s algorithm. Details can be found in [16].

SAB successively narrows the probability bound of achieving or failing some criteria by iteratively *splitting* regions comprising a partition of the possible input values *and* then *bounding* both the possible behaviors within the smaller regions and the probabilities of being in those smaller regions (using the input probability density bound). SAB marks the regions whose possibilities always satisfy or fail the criteria.

Figure 2 shows two examples of splitting. In the one marked *Behavior*, the criterion is $ab < 3$, and the original region is $a, b \in [0, 2]$. In this region $ab \in [0, 4]$, so it sometimes passes and sometimes fails the criterion. Split⁷ the region along $a = 1$ into the two sub-regions X and Y . In X , $a \in [0, 1]$, so $ab \in [0, 2]$. Because the criterion is always satisfied, mark X . In contrast, $a \in [1, 2]$ in Y , which means $ab \in [0, 4]$, so Y is not marked.

In the example marked *Probability*, $\alpha = (c \in [0, 1])$ is the original region and $f(c)$ is a lower bound on probability density at c . SAB finds the lower bound on the probability of being in α of $\Pr(\alpha) > 0.5$. This bound is the sum of areas q and

⁷At present, SAB splits a region by bisecting it.

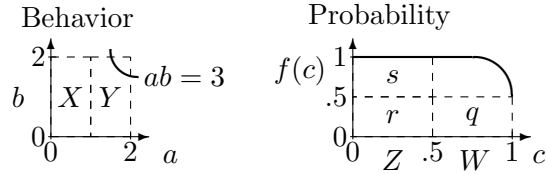


Figure 2: Examples of Splitting

r , found by multiplying 1, α 's length, by 0.5, the lowest value of $f(c)$ in α .⁸ Split the region at $c = 0.5$ into the two sub-regions Z and W . By a method similar to the one above, SAB finds a lower bound on $\Pr(Z)$ of 0.5 (sum areas r and s), and a lower bound on $\Pr(W)$ of 0.25 (area q). Sum the lower bounds of $\Pr(Z)$ and $\Pr(W)$ to get a new lower bound of 0.75 on $\Pr(\alpha)$.

As hinted by these two examples, *as long as* the bounding method used tends to reduce the range of possibilities as a region of input values gets smaller, this continued splitting will mark more and more of the interval of all possible input values. And *as long as* the bounding method tends to reduce the gap between a density bound's upper and lower bound⁹ in a region as the region gets smaller, the bound on the probability of being in a marked region will improve.

To find a lower bound on $\Pr(\text{satisfy criteria})$ sum the lower probability bounds of all the regions marked as satisfying the criteria. Similarly, one can find a lower bound on $\Pr(\text{fail criteria})$. One minus the latter is an upper bound on $\Pr(\text{satisfy criteria})$.

⁸Better methods of bounding probabilities are described in [16].

⁹Yes, we are bounding a bound here.

5 SAB: *PVR* Example Revisited

This section re-examines the *pvr* example when using the Gaussian density as a “bound” (first density bound) and SAB as the bounding method. To bound $\Pr(pvr > 1.62)$, SAB looked at the interval of all possible inputs¹⁰ (given to SAB as one region):

$$pap \in [1.0, 88.0], lap \in [1.0, 88.0], co \in [1.0, 100].$$

A lower bound on *pvr*, written $lb(pvr)$, is

$$\max(0, [lb(pap) - ub(lap)]/ub(co)) = 0,$$

and an upper bound ($ub(pvr)$) is

$$[ub(pap) - lb(lap)]/lb(co) = 87.0.$$

pvr can be either greater or less than 1.62, so SAB split the region in two along the *co* dimension:

$$subspace1 : \quad pap \in [1.0, 88.0], lap \in [1.0, 88.0], co \in [1.0, 50.5]$$

$$subspace2 : \quad pap \in [1.0, 88.0], lap \in [1.0, 88.0], co \in [50.5, 100.0]$$

SAB then checked and split as appropriate. Regions like

$$pap \in [20.75, 25.47], lap \in [15.95, 17.32], co \in [6.41, 7.19], (pvr \in [0.756, 1.484])$$

¹⁰Note that the given range of possible values is wider than necessary. For example, one could tighten them to $co \leq 30$ and $lap \leq 45$. These wider bounds will not affect the correctness of the results. They will slow SAB down, but probably not by much: the inputs should have little (if things are slightly inconsistent) or no chance of being in the extra area included by the wider bounds, and SAB concentrates first on the regions of the possible input values that have the highest estimated probability of occurring. So when in doubt about the bounds, err on the side of including something that cannot occur.

where pvr is either always $>$, or ≤ 1.62 , were marked. SAB found lower bounds on the probabilities of being in these marked regions (the one above has a probability ≥ 0.002).

As SAB recursively splits and checks regions, it tightens the probability bound for satisfying the criteria. When the bound is tight enough, or SAB runs out of time or another resource, it can be stopped. In this example, when SAB was stopped, it gave a lower bound of 0.042 on the probability of being in a passing region (one where $pvr > 1.62$), and 0.438 for a failing region ($pvr \leq 1.62$). If a tighter bound was desired, one could have restarted SAB with the then current set of regions. Since this joint density bound includes all of the probability mass, SAB can, barring round-off error in the floating point math, get the bound to be arbitrarily tight if given enough computing time. In general, if a joint density bound includes $n \times 100\%$ of the probability mass, SAB can, barring round-off error, get the bound to have a gap of $1.0 - n$ between the upper and lower figure. So if a lower density bound includes 70% of the probability mass, the tightest bound SAB could give on the chances of passing some criteria would have a gap of 0.3 between the lower and upper figures (such as a lower bound of 0.6 and an upper bound of 0.9).

6 Problems with SAB

SAB has two difficulties. First, as the number of inputs increases, the complexity of the density bound often increases to beyond SAB's capacity. In fact, SAB is not fast for small problems either. The examples with SAB in Section 3 each took 30 to 60 minutes (1 to 10 thousand iterations) to run on a Symbolics 3640 (or 3650)

Lisp machine.¹¹

SAB's second problem is that it can make little use of upper density bounds. This is because SAB derives its bounds by summing the bounds of the sub-regions. On a lower input density bound, SAB sums up lower probability bounds, and on an upper density bound, SAB sums up upper probability bounds. If a few of the upper sub-region probability bounds being summed are loose, say close to one, then the entire sum will be close to one or more (since all the probability bounds are at least zero), which will make for a loose bound. With lower bounds on the other hand, a few loose lower sub-region probability bounds, say close to zero, will not affect the sum very much. For example, let there be 5 sub-regions, each with a probability of 0.1 of containing the measured input values. Then the chances of event α , that the measured values are in one of the sub-regions, is $5(0.1) = 0.5$. If the upper bound on the chances of each of four of these sub-regions is 0.1, and is 0.9 in the fifth sub-region, then an upper bound on $\Pr(\alpha)$ is $4(0.1) + 0.9 \geq 1$, a loose bound. On the other hand, if the lower bound on the chances of each of four of these sub-regions is 0.1, and is 0.0 in the fifth one, then a lower bound on $\Pr(\alpha)$ is $4(0.1) = 0.4$, which is still close to the actual chance of 0.5.

Unfortunately for SAB, if one gives an input density bound where a mean or some other parameter can vary over a wide range, the upper bound becomes much more important than the lower one. This is because as some parameter varies over a wider range, the intersection of the densities with all the different allowable parameter values gets small, so the lower bound covers little (is loose). For example,

¹¹Part of the reason for the slowness was that the small examples were run with all the bounding machinery needed for the larger problems. No attempt was made to optimize. LISP was used because of its storage management, which was needed to easily handle the newly split regions.

in the “*vary mean, LB*” bound in Figure 1, if the mean is allowed to vary more, the two extreme Gaussian curves depicted will move farther away from each other, and the area in common to them (the lower bound) will shrink. When a lower bound covers little, most of the probability mass will not be covered by the bound and can go anywhere. At this point, one will not be able to infer much on the chances of any input values occurring. An example of a lower bound covering little is one which only has 0.05 (5%) of the probability mass under its curve. If one uses this bound to bound the chances of some event α , one will get a lower bound on $\Pr(\alpha)$ of x and a lower bound on $\Pr(\neg\alpha)$ of $0.05 - x$, where x is a number between 0.0 and 0.05. An upper bound on $\Pr(\alpha)$ is one minus a lower bound on $\Pr(\neg\alpha)$, so the upper bound is $1 - (0.05 - x) = 0.95 + x$. Then the lower bound on $\Pr(\alpha)$ is at most 5% while the upper bound is at least 95%, which does not tell one much about $\Pr(\alpha)$.

While a loose lower density bound is useless, a loose upper density bound can still yield useful probability upper bounds. This is because there may be large regions of possible input values which are known to be very unlikely to occur.

7 HMC (Method 2)

These problems with SAB led to trying to apply Monte Carlo (MC) techniques in conjunction with density upper bounds. As mentioned in Section 2, the common MC techniques are inadequate. This section describes HMC, a hit-or-miss version of sample-mean Monte Carlo which applies to the problems of interest. The first subsection describes the original sample-mean MC and why it is inadequate for the problems of interest. The next subsection describes how to convert sample-mean MC into HMC. The last subsection gives an example of a large problem to which

HMC was applied.

7.1 Sample-Mean Monte Carlo

Sample-mean Monte Carlo (called *crude Monte Carlo* in [5]) estimates the area under a curve c within some region (interval) as follows:

1. Randomly sample the curve in the region of interest: first take some random samples of sets of input values in that region, and then for each set of values, \underline{x}_i , find $c(\underline{x}_i)$, the value of the curve at \underline{x}_i .
2. Estimate the average value of the curve in the region by finding the average value of the $c(\underline{x}_i)$ s: $A_c = \sum_i c(\underline{x}_i)/n$, where n is the number of samples. The sample standard deviation of the $c(\underline{x}_i)$ s is

$$s = \sqrt{\sum_i (c(\underline{x}_i) - A_c)^2 / (n - 1)}.$$

The standard deviation of A_c is estimated by $s_A = s/\sqrt{n}$.

3. Multiply A_c by the size or volume S of the region or interval: $A_c \times S$. This result estimates the area under the curve within the region of interest. The standard error of this estimate is $s_A \times S$. Since S is constant and $s_A \propto 1/\sqrt{n}$, the estimate becomes more accurate as the number of samples increase. However, since the proportionality is to $1/\sqrt{n}$ and not to $1/n$, each additional sample added has less effect on the accuracy.

As previously mentioned, sample-mean MC is an MC integration technique. Such techniques have the disadvantage that one has to find all the regions of possible input values that satisfy the criteria, as these regions are the ones to be integrated over. As an illustration of this, consider the *pvr* example in Section 3. The interval

of all possible input values is the region defined by $pap \in [10, 80]$, $lap \in [2, 45]$, and $co \in [1, 30]$. To find a bound on $\Pr(pvr > 1.62)$, one needs to integrate over the density bound in all the regions of possible input values where the criterion of $1.62 < pvr = (pap - lap)/co$ is satisfied. Two of these regions are

$$pap \in [19, 80], lap \in [2, 15], co \in [1, 2]$$

and

$$pap \in [19, 80], lap \in [2, 11], co \in [2, 4].$$

Finding all such regions in the interval of all possible input values is difficult.

7.2 From Mean-Sample MC to HMC

Although it is hard to find all the regions of input values that satisfy the criteria, one can easily tell if a given set of values do so. In the pvr example, if one were given the input values of $pap = 80$, $lap = 30$, and $co = 5$, one could substitute them into the equation $(pap - lap)/co$ to find that the corresponding pvr is 10, which passes the criterion.

Another observation is the following. Suppose one defines a new curve $c^*(\underline{x})$:

$$c^*(\underline{x}) = \begin{cases} c(\underline{x}) & \text{if } \underline{x} \text{ satisfies the criteria} \\ 0 & \text{otherwise} \end{cases}$$

where $c(\underline{x})$ is the original curve and \underline{x} can take on all possible input values. Then the area under c^* within the region of all possible input values is equivalent to the area under c within the regions of values that satisfy the criteria.

One can use these two ideas to modify sample-mean MC to obtain HMC by substituting all occurrences of $c(\underline{x}_i)$ s with $c^*(\underline{x}_i)$ s in the three-step algorithm described in Section 7.1. To obtain $c^*(\underline{x}_i)$ for each sample, \underline{x}_i , see if \underline{x}_i satisfies the

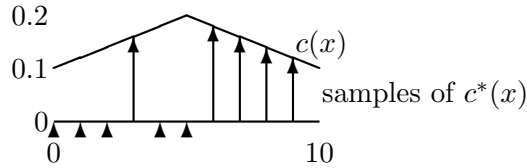


Figure 3: HMC Example

criteria. If it does, find $c(\underline{x}_i)$; otherwise use 0. When the \underline{x}_i s come from the interval of all possible input values, the result $A_c \times S$ in the last step estimates the area under c which satisfies the criteria.¹² When c is a density upper bound, the result estimates an upper bound on the probability of satisfying the criteria. Similarly, when c is a density lower bound, the result estimates a lower probability bound.

An example of using this routine is as follows: let x range from 0 to 10 (so $S = 10$), let the criteria be $q(x) = x^3/3 - 4x^2 + 15x \geq 18$, and let the density upper bound, $c(x)$, be $0.02(10 - |x - 5|)$. The modified density bound is

$$c^*(x) = \begin{cases} c(x) & \text{if } q(x) \geq 18 \\ 0 & \text{otherwise} \end{cases}$$

Let the sample-mean routine pick 10 independent random samples of x .¹³ Then the resulting samples are as shown in Table 2 and Figure 3.

An estimate of the area under the upper density bound is 1.5, which is derived by multiplying the size S , by 0.15, the average of the sample c values. The estimated standard deviation on this estimate is 0.1, which is derived by multiplying S , by 0.01, the sample standard deviation for the average of c . The actual area equals the estimate of 1.5.

¹²HMC, the resulting algorithm, is related to conditional Monte Carlo [5, Ch. 6] [4, p. 12]. HMC may be a type of the rejection technique described on page 12 of Halton [4].

¹³To simplify the example, the “independent random” samples “happen” to be the integers between 0 and 9.

To get an estimate of the upper bound on $\Pr(q(x) \geq 18)$ and a guess of the standard deviation on this estimate, multiply the corresponding summary figures for c^* by S . The result is $\Pr(q(x) \geq 18) \leq 0.76$ with a standard error of about 0.26. The actual upper bound is 0.56, which is less than one standard error from the estimate.

7.3 Summary of a Larger Example

For a larger example, I ran HMC on a 30⁺-variable, 25⁺-equation cardiovascular model that includes Equation 1 [8, 9].¹⁴ Using data on mitral stenosis patients [2], an upper density bound for the six input variables was constructed. It basically had the form of a joint Gaussian density, but the mean blood pressure was allowed to be anywhere in the interval between 85.7 and 89.7, and the mean heart rate, in the interval between 80.9 and 84.9. The bound was formed by finding the maximum possible value for the density at each sample point by using some of the bounding techniques in Section 8. This density bound was used with HMC to bound the chances of both $pvr \leq 5.0$ and $svr \geq 40.0$ occurring in a population of mitral stenosis patients. The variable svr is the systemic vascular resistance, or general body resistance to blood flow. The technique found that the upper bound on the probability of satisfying this criteria is less than 0.0085 (with a standard deviation of 0.0012 on the figure). Unfortunately, getting these results took over 100,000 sample points (4000 minutes on a Symbolics 3640 Lisp machine).¹⁵

¹⁴The model in the references used several piecewise linear equations. In this example they were replaced with ones which used exponentiation to constant powers.

¹⁵It took a long time to run each sample point because testing each sample on the constraints involved solving nonlinear simultaneous equations. These were solved using the bounding mechanism to be described in the next section. The mechanism is slow and does not solve all possibly

8 Finding Numeric Bounds

Many of SAB's parts need expressions to be bounded. For expressions with algebraic, logarithmic, and exponential functions (the type in the models to be used), perfect bounding algorithms have not been built. The type of algorithm used here will find bounds that indicate what is truly unachievable,¹⁶ but those bounds may not be the tightest possible. For example, having a program find that $x < 7$, when in fact, it can be shown that $x < 3$. I have implemented an augmented version of bounds propagation [12], which does the following interval arithmetic [11]:

- Bound an operation's result using bounds on the operands. For example:
$$ub(a + b) \leq ub(a) + ub(b).$$
- Bound an operand using bounds on an operation's result and the other operands. For example: $ub(a) \leq ub(a + b) - lb(b)$.

The "binder" examines expressions and updates bounds with these operations. It *iterates* over the expressions until every one that might produce a change has been examined at least once and all the recent bound changes are below a certain threshold.

solvable sets of equations. But because of the type of bounding the mechanism does, if there are multiple solutions to the equations, the mechanism will warn about the possibility and give an interval in which all the solutions are located. Most equation solvers will just present the first solution they find. Using standard numeric techniques like Newton-Raphson will save time.

¹⁶In practice, the accuracy of this may be limited by round-off error.

9 Discussion

This paper describes two methods which use a bound on the joint density of the inputs to bound the likelihood of some possible behavior. The first is called SAB. It analytically bounds the likelihood, narrowing the bound as more iterations are permitted. However, SAB can only use lower-bound information on the joint density. This information is less useful than upper-bound information. The second method is a hit-or-miss version of sample-mean Monte Carlo technique called HMC. It only finds approximate likelihood bounds – but the accuracy increases with the number of samples taken – and it gives a measure of the error magnitude. Also, unlike SAB, HMC can handle both upper and lower density bounds, and can handle larger problems than SAB can.

Future work on SAB itself includes testing how large a problem it can handle and expanding it to more quickly bound a variable’s mean, variance, median, 90% confidence interval, etc. I will also explore splitting a region at the selected variable’s median value (or some approximation) within the region. This can handle infinite intervals (bisection cannot), which permits an initial region where each variable is within the all-inclusive range of $[-\infty, \infty]$.

Future work on HMC includes employing it to run the cardiovascular model mentioned in Section 7.3 in order to examine the effects of heart disease therapies, and trying variance reduction techniques for MC algorithms (like stratified sampling) to speed up HMC. To speed-up instances of solving simultaneous equations, I may use more standard numeric programming methods such as Newton-Raphson.

At present, both algorithms are slow, so it would be useful to generate preliminary results by applying some fast method like moments [3], which just need to solve a few equations (without need for iterations) to give an estimate.

On matters other than algorithms, work needs to be done on finding the types of density bounds that are the most common, easiest to specify, and most useful. Candidates for easy-to-specify bounds are common densities with bounded parameters. An example is a Gaussian density with a mean between 0 and 1. One can generate such bounds by using information from parameter confidence intervals.

Despite uncertainty in input density shapes and parameter values, bounds on input densities have not really been utilized to bound the chances of events. This work describes two bounding methods which have the features of being able to handle events beyond the ones in a pre-enumerated list, estimating the amount of error in an answer, and giving better answers as more iterations or samples are allowed.

Acknowledgments

This research was supported in part by the National Heart, Lung, and Blood Institute through grant R01-HL33041. The idea for SAB arose after a thought-provoking discussion with Mike Wellman, who also, along with Peter Szolovits, William Long, Alvin Drake, Peter Kempthorne, and Peter Huber, have reviewed parts of this paper.

References

- [1] Daniel G. Bobrow, editor. *Qualitative Reasoning about Physical Systems*. MIT Press, 1985.
- [2] G. Giuffrida, G. Bonzani, S. Betocchi, F. Piscione, P. Giudice, D. Miceli, F. Mazza, and M. Condorelli. Hemodynamic response to exercise after pro-

- pranolol in patients with mitral stenosis. *The American Journal of Cardiology*, 44:1076–1082, November 1979.
- [3] G. Hahn and S. Shapiro. *Statistical Models in Engineering*. John Wiley & Sons, Inc., 1967.
- [4] John H. Halton. A retrospective and prospective survey of the monte carlo method. *SIAM Review*, 12(1):1–63, 1970.
- [5] J. M. Hammersley and D. C. Handscomb. *Monte Carlo Methods*. Barnes and Noble, Inc., 1965.
- [6] Eric J. Horvitz, David E. Heckerman, and Curtis P. Langlotz. A framework for comparing alternative formalisms for plausible reasoning. In *Proceedings of the National Conference on Artificial Intelligence*, pages 210–214. American Association for Artificial Intelligence, 1986.
- [7] Mark Johnson. *Multivariate Statistical Simulation*. John Wiley and Sons, New York, 1987.
- [8] W. J. Long, S. Naimi, M. G. Criscitiello, and R. Jayes. Using a physiological model for prediction of therapy effects in heart disease. In *Proc. of the Computers in Cardiology Conf.* IEEE, October 1986.
- [9] William J. Long, Shapur Naimi, M. G. Criscitiello, and Robert Jayes. The development and use of a causal model for reasoning about heart failure. In *Symposium on Computer Applications in Medical Care*, pages 30–36. IEEE, November 1987.
- [10] Dean T. Mason. *Congestive Heart Failure, Mechanisms, Evaluation, and Treatment*. Dun-Donnelley, New York, 1976.

- [11] H. Ratschek and J. Rokne. *Computer Methods for the Range of Functions*. Halsted Press: a division of John Wiley and Sons, New York, 1984.
- [12] Elisha P. Sacks. Hierarchical reasoning about inequalities. In *Proceedings of the National Conference on Artificial Intelligence*, pages 649–654. American Association for Artificial Intelligence, 1987.
- [13] William C. Shoemaker. Physiology, monitoring and therapy of critically ill general surgical patients. In William C. Shoemaker and Edward Abraham, editors, *Diagnostic Methods in Critical Care*, chapter 4, pages 47–86. Marcel Dekker, Inc., New York, 1987.
- [14] Peter Szolovits and Stephen G. Pauker. Categorical and probabilistic reasoning in medical diagnosis. *Artificial Intelligence*, 11:115–144, 1978.
- [15] Brian C. Williams. MINIMA: A symbolic approach to qualitative algebraic reasoning. In *Proceedings of the National Conference on Artificial Intelligence*, pages 264–269. American Association for Artificial Intelligence, August 1988.
- [16] Alexander S. Yeh. Predicting the likely behaviors of continuous nonlinear systems in equilibrium. In *The Fourth Workshop on Uncertainty in Artificial Intelligence*, pages 374–381. Advanced Decision Systems, Mountain View, CA, August 1988.

NAME	MEAN	STD DEV	Correlation Coef.		
			pap	lap	co
pap	23.94	3.38	1.0	.861	.096
lap	15.29	3.08	.861	1.0	-0.044
co	6.49	1.20	.096	-0.044	1.0

Table 1: *PVR* Example

x	0	1	2	3	4	5	6	7	8	9	A_c	s_A
$c(x)$	0.10	0.12	0.14	0.16	0.18	0.20	0.18	0.16	0.14	0.12	0.15	0.01
$c^*(x)$	0.0	0.0	0.0	0.16	0.0	0.0	0.18	0.16	0.14	0.12	0.076	0.026
$q(x)$	0.0	11.3	16.7	18.0	17.3	16.7	18.0	23.3	34.7	54.0		

Table 2: HMC Example Samples