

# CALIBRATION OF UNCERTAIN INPUTS TO COMPUTER MODELS USING EXPERIMENTALLY MEASURED QUANTITIES AND THE BMARS EMULATOR

**H.F. Stripling and R.G. McClarren**

Texas A&M University  
Department of Nuclear Engineering  
3133 TAMU; College Station, TX 77843-3133  
h.stripling@tamu.edu; rgm@tamu.edu

**C.C. Kuranz, M.J. Grosskopf, E. Rutter and B.R. Torralva**

University of Michigan  
Atmospheric, Oceanic, and Space Science Department  
2455 Hayward; Ann Arbor, MI 48109-2104  
ckuranz@umich.edu; mikegros@umich.edu; ruttere@umich.edu; bentorra@umich.edu

## ABSTRACT

We present a method for calibrating the uncertain inputs to a computer model using available experimental data. The goal of the procedure is to produce posterior distributions of the uncertain inputs such that when samples from the posteriors are used as inputs to future model runs, the model is more likely to replicate (or predict) the experimental response. The calibration is performed by sampling the space of the uncertain inputs, using the computer model (or, more likely, an emulator for the computer model) to assign weights to the samples, and applying the weights to produce the posterior distributions and generate predictions of new experiments within confidence bounds. The method is similar to the Markov chain Monte Carlo (MCMC) calibration methods with independent sampling with the exception that we generate samples beforehand and replace the candidate acceptance routine with a weighting scheme.

We apply our method to the calibration of a Hyades 2D model of laser energy deposition in beryllium. We employ a Bayesian Multivariate Adaptive Regression Splines (BMARS) emulator as a surrogate for Hyades 2D. We treat a range of uncertainties in our system, including uncertainties in the experimental inputs, experimental measurement error, and systematic experimental timing errors. The results of the calibration are posterior distributions that both agree with intuition and improve the accuracy and decrease the uncertainty in experimental predictions.

*Key Words:* Calibration, Uncertainty Quantification, Predictive Science, Bayesian MARS

## 1. INTRODUCTION

We are interested in calibrating the uncertain inputs to a computer model such that the computer model is more likely to replicate experimental results. In essence, we seek to improve the accuracy of a model equation relating simulation to reality by “learning” a posterior distribution of uncertain parameters. The model equation for this work assumes that an experimentally

measured quantity  $Y$  can be related to a simulation  $f(\cdot)$  of that quantity without a model discrepancy function:

$$Y_{\text{meas}}(\vec{x}_i) = F(\vec{x}_i) + \epsilon = f(\vec{x}_i, \vec{\theta}) \quad (1)$$

In Eq. (1),  $\vec{x}_i \in \mathbb{R}^d$  is a vector of independent or experimental inputs corresponding to experiment  $i$ ,  $\vec{\theta} \in \mathbb{R}^p$  is a vector of inputs to the computer model whose values are uncertain,  $F(\cdot) = Y_{\text{true}}(\vec{x}_i)$  represents the true experimental response for input settings  $i$ , and  $\epsilon$  is an error term accounting for stochastic behavior of nature, unknown unknowns, or experimental measurement error. The calibration routine requires access to the simulator response at-will, which is not computationally feasible for any calculation of practical importance. Therefore we employ an emulator  $\eta(\cdot)$  to generate predictions of the simulator response. We account for regression error of the emulator,  $\zeta$ , by rewriting Eq. (1):

$$F(\vec{x}_i) = \eta(\vec{x}_i, \vec{\theta}_i) + \zeta - \epsilon \quad (2)$$

The error terms  $\zeta$  and  $\epsilon$  are typically well characterized. In our application, the regression error  $\zeta$  is estimated during the construction of the emulator, and  $\epsilon$  can be calculated or estimated heuristically based on knowledge of the physical system and measurement equipment or instrumentation. Application of Eq. (2) becomes more interesting if we treat each term as a *distribution* instead of a single realization. The emulator term becomes a distribution if it is evaluated at a distribution of  $\vec{\theta}$ , and the result is a distribution of predictions for the true experimental response  $F(\cdot)$ .

The work presented here focuses on calibrating the uncertain inputs  $\vec{\theta}$  to a posterior distribution such that the predictions of  $F(\cdot)$  are made more accurately and with less uncertainty. In the following sections, we outline our BMARS emulator (Bayesian Multivariate Adaptive Regression Splines), detail the procedure for generating weights for the uncertain inputs, and present results of leave-one-out type predictions as well as predictions of entirely new experiments. Our application is related to the CRASH (Center for Radiative Shock-Hydrodynamics) PSAAP (Predictive Science Academic Alliance Program) problem; we use data from experimental measurements of laser breakout time to calibrate three uncertain inputs to the Hyades 2D computer model.

## 2. DESCRIPTION OF THE METHODS AND THE APPLICATION OF INTEREST

### 2.1 The Emulator: Bayesian Multivariate Adaptive Regression Splines (BMARS)

The MARS algorithm is a partition-based curve-fitting technique which attempts to emulate the mapping between a function's inputs and outputs as a summation of spline functions. Multivariate spline functions are simply products of one-dimensional spline functions; these 1D spline functions are defined to be zero on part of the domain and a polynomial of some order on the remainder of the domain. The knot of the spline is the coordinate at which this definition changes, and the direction of the spline describes whether the non-zero portion of the spline

is in the positive or negative direction from the knot. In our implementation, knot points are restricted to the coordinates of the training data given to the model.

Given a set of input or training data, the classical formulation[3] uses a semi-stochastic method to generate a basis function of the form:

$$B(x) = \beta_0 + \sum_{k=1}^{\mathbf{K}} \beta_k \prod_{l=0}^{\mathbf{I}} (x_l - t_{k,l})_+^{o_k} \quad (3)$$

where  $\vec{x}$  is a vector of inputs,  $t_{k,l}$  is the knot point in the  $l^{th}$  dimension of the  $k^{th}$  component, the function  $(y)_+$  evaluates to  $y$  if  $y > 0$ , else it is 0,  $o$  is the polynomial degree of the  $k^{th}$  component,  $\beta_k$  is the coefficient of the  $k^{th}$  component,  $\mathbf{K}$  is the maximum number of components of the basis function, and  $\mathbf{I}$  is the maximum allowed number of interactions between the  $L$  dimensions of the input space. Note that the formulation does not require that each of the  $k$  components have a term in each dimension of  $\vec{x}$ . Also, the function has only a maximum of  $\mathbf{K}$  components, but implementations typically search for the optimal fit while minimizing  $k$ .

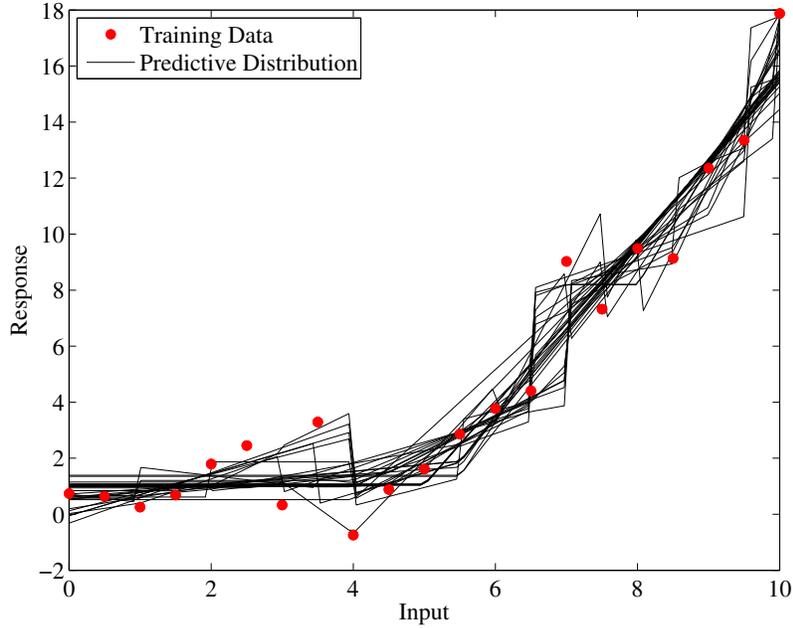
The Bayesian extension of MARS (hereby named BMARS)[2] generates an ensemble of predictive MARS functions such that an evaluation generates a predictive distribution instead of a single prediction. Step zero of the algorithm generates a classical basis function of the form (3). A Markovian process proposes a change to the current model in the form of an addition, deletion, or modification of a spline. When a new spline is created, the algorithm randomly chooses its order, knot point, direction, and level of interaction. The algorithm iterates this random selection process and accepts/rejects proposal basis functions based on a likelihood calculation. This likelihood is a function of the candidate's fit to the training data and the number of splines in its basis function. The coefficients,  $\beta$ , are found via matrix inversion, but are perturbed on a noise distribution that is updated as the process evolves. Similarly, the error in the regression ( $\zeta$  in Eq. (2)) is estimated at each iteration based on the current model's approximation of the training data.

We illustrate the utility of this emulator in Fig. 1, where we show a one-dimensional BMARS fit to piecewise-linear training data perturbed by a  $N(0,1.5^2)$  error term. The classical MARS algorithm would find just one of these predictive functions, while the Bayesian extension provides a distribution of predictions generated from samples from the posterior MCMC realizations.

## 2.2 Calibrating the Uncertain Inputs via Sampling and Filtering

The goal of calibration is to find values or distributions of an approximate model's inputs such that the approximate model becomes a more accurate predictor of reality. As we outlined in the introduction, an experiment and its approximate model typically share a set of inputs,  $\vec{x}$ . The simulation will also have a number of other inputs,  $\vec{\theta}$ , which can take the form of non-physical tuning parameters or other physical inputs which are not experimentally varied. We seek distributions of these uncertain parameters such that the simulation produces a more accurate prediction when  $\vec{x}_{\text{sim}} = \vec{x}_{\text{exp}}$ .

We propose a two-step calibration procedure. Step one is to generate a BMARS model for the simulator,  $f(\cdot) = B(\cdot) + \zeta$ . The model will be a function of  $d+p$  variables; that is, we make no



**Figure 1: An example 1D BMARS fit to piecewise-linear data perturbed by a noise term.**

distinction between independent and uncertain inputs in step one. Determining the adequacy of the emulator is an independent subject; we note that the following analysis operates under the assumption that the emulator is an adequate representation of the simulator, or at least that it appropriately estimates the regression error  $\zeta$ .

Step two relates the BMARS emulator to a number of available experimental measurements at inputs  $\{x_i\}_{i=1}^I$ . The calibration procedure is as follows:

1. Generate  $N$  samples of the uncertain input space. Samples should be contained within the convex hull of the available simulation runs to avoid extrapolating with the emulator and should be as dense as possible. At this time, we have only considered uniform sampling.
2. for  $i = 1 \dots I$  and  $n = 1 \dots N$ 
  - (a) Randomly choose  $M$  indices from the posterior distribution of the BMARS model;
  - (b) Generate and normalize a discrete probability distribution function,  $g\left(\eta(\vec{x}_i, \vec{\theta}_n)\right)$  by evaluating  $\eta_m(\cdot) = B_m(\cdot) + \zeta_m$ ,  $m = 1 \dots M$ ;
  - (c) Compute a local weight,  $\omega_{i,n} = L(Y_{\text{meas}}(x_i)|g, \epsilon)$ , which should represent some interpretation of the uncertainties in the problem.
3. Compute the global weight,  $w_n = \prod_{i=1}^I \omega_{i,n}$ .

The result of the algorithm is a global (normalized) weight assigned to each of the  $N$  samples of the uncertain input space. The weights are computed via the likelihood evaluation in step 2.(c), which should be tailored to information in the specific problem. Below we discuss a number of possible likelihood models which could be employed for different problems.

The weights for each sample should be proportional to the likelihood that the simulator will replicate experimental results if it is run at that sample. Because the global weight is a product of the local weights, the samples with the largest global weights will be those that most consistently approximate each of the  $I$  experimental calibration points. We represent a posterior distribution of a given dimension of  $\vec{\theta}$  by simply binning the weights of the discrete samples in that dimension.

We apply the weights to generate predictions and confidence intervals for new experiments by computing weighted means and standard deviations of BMARS evaluations at the desired experimental input. The expected value of an experiment is simply

$$\bar{Y}_{\text{pred}}(x_i) = \frac{1}{M} \sum_{m=1}^M \sum_{n=1}^N \eta_m(x_i, \vec{\theta}_n) w_n.$$

We also use the global weights to determine confidence bounds. The weighted percentile associated with the prediction  $\eta_m(x_i, \vec{\theta}_n)$  is computed as

$$p_{n,m} = \frac{100}{S_N} \left( S_n - \frac{w_n}{2} \right)$$

where  $S_n$  is the partial sum  $\sum_{i=1}^n w_i$ . Then, to find the predictive value associated with the  $p^{\text{th}}$  percentile, we simply use linear interpolation:

$$Y_{\text{pred},m}(p) = Y_{\text{pred},m}^{(k)} + \frac{p - p^{(k)}}{p^{(k+1)} - p^{(k)}} \left( Y_{\text{pred},m}^{(k+1)} - Y_{\text{pred},m}^{(k)} \right)$$

where  $k$  is an integer such that  $p^{(k)} \leq p \leq p^{(k+1)}$ . For example, to compute a 90% confidence interval, we find  $Y_{\text{pred},m}(5\%)$  and  $Y_{\text{pred},m}(95\%)$  for all  $m$  and then average over  $m$ . Again, here  $m$  is the randomly chosen indices from the posterior distribution of the BMARS model.

Step 2.(c) of the algorithm above will require interpretation of the specific application; that is, the modeler must choose which uncertainties to include in the analysis and how to properly account for them in the likelihood distribution  $L(\cdot)$ . At a minimum the system will have uncertainty from emulator regression and experimental measurement. In the case of BMARS, the regression error is estimated as a normal distribution,  $\zeta \sim N(0, \chi^2)$ . If the measurement error is also estimated as normal ( $\epsilon \sim N(0, \tau^2)$ ), then the function  $L$  is equivalent to evaluating the normal pdf:

$$L(\vec{\theta}_{i,n}) \propto N \left( Y_{\text{meas}}(x_i) \left| \mu = \frac{1}{M} \sum_{m=1}^M B_m(\vec{x}_i, \vec{\theta}_n) + N(0, \chi_m^2), \sigma^2 = \chi^2 + \tau^2 \right. \right) \quad (4)$$

Other sources of uncertainty or distributions of those uncertainties will require the modeler to generate a tailored likelihood function. For example, the experimental measurement data in our application is subject to a systematic, uniform timing error of appreciable magnitude. We attempt to account for this uncertainty first by including a representative variance term in a likelihood calculation similar to Eq. (4), and then propose to construct a likelihood distribution which is a summation of the measurement, systematic, and regression error distributions.

### 2.3 The Application: Calibrating Hyades 2D to Experimentally Measured Shock Breakout Times

In this paper we present results of calibrating the Lagrangian rad-hydro code Hyades 2D (H2D) to experimentally measured shock breakout times. The application is of interest to the Center for Radiative Shock Hydrodynamics (CRASH), comprised of researchers from the University of Michigan and Texas A&M University. Specifically, the center is developing a model for laser-driven shock propagation through a xenon-filled tube. The model is currently initialized by the energy deposition calculations from Hyades 2D, and a fundamental quantity of interest is the time required for the laser-driven shock wave to propagate through a disk of beryllium at one end of the tube (this is the shock breakout time).

The independent experimental variables,  $\vec{x}$ , are the beryllium disk thickness and laser energy. H2D shares these variables, and we are including three other H2D inputs as the uncertain parameters ( $\vec{\theta}$ ): beryllium gas gamma constant, a flux limiter, and tube-wall opacity. We have results from 8 experiments, wherein each experiment the shock breakout time was measured by three different diagnostics (called asbo1, asbo2, and sop). These results are give in Table I. We also have a total of 104 H2D runs generated from latin hypercube sampling of the 5 inputs. The inputs to these H2D runs are summarized in Table II.

**Table I: Available experimental data for H2D calibration.**

Experiment	Be Disk	Laser	asbo1 (ps)	asbo2 (ps)	sop (ps)
	Thickness ( $\mu m$ )	Energy (J)			
1	21	3837.6	504	486	462
2	20	3925.2	467	475	430
3	20	3937.6	437	450	none <sup>a</sup>
4	19	3887.8	419	410	436
5	20	3914.6	425	467	456
6	20	3912.8	442	476	470
7	19	3923.3	447	456	470
8	19	3945.8	410	417	418

<sup>a</sup>No sop data recorded for this shot

## 3. RESULTS OF CALIBRATION APPLIED TO THE HYADES 2D UNCERTAIN INPUTS

### 3.1 Calibration of a Simplified Problem

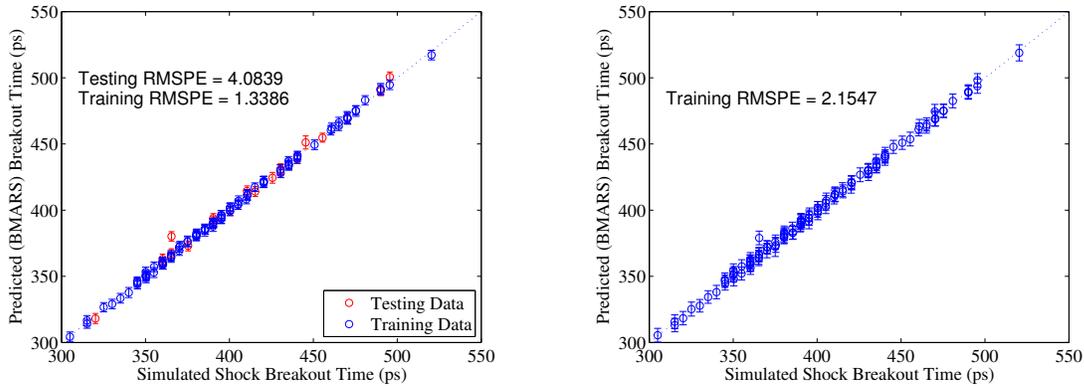
As a simple proof-of-concept for the method, we first performed the calibration in a simplified setting. The goal will be to perform accurate leave-one-out predictions, where we perform the calibration on seven of the eight experimental results and (in turn) try to predict the eighth.

**Table II: Hyades 2D simulation input ranges and distributions.**

Input	Distribution
Laser Energy (J)	U[3610,3990]
Disk Thickness ( $\mu m$ )	U[18,22]
Be gamma	U[1.4,1.75]
Wall Opacity	U[0.7,1.3]
Flux Limiter	U[0.05,0.075]

We simplify the problem by considering only the asbo1 measurements, which have an associated measurement error term  $\epsilon \sim N(0, 10\text{ps}^2)$ . We attempt to calibrate the three uncertain inputs: beryllium gamma, wall opacity, and flux limiter.

Following the outlined procedure, we first generate a BMARS fit to the available H2D data. As we briefly mentioned, an accurate emulator is an underlying assumption and necessity of this procedure. We tuned and validated the emulator fit by performing a series of tests in which we fit a BMARS model to just 84 of the 104 available simulation runs and used the model to predict the remaining 20 runs. We found the BMARS emulator to have a consistent root-mean-square predictive error of  $\sim 1\%$  in these predictions, which we considered sufficient. As a final step, we generated a BMARS fit to all 104 H2D simulations and used this model for the calibration. Figures 2(a) and 2(b) illustrate an example validation fit and the final fit, respectively.



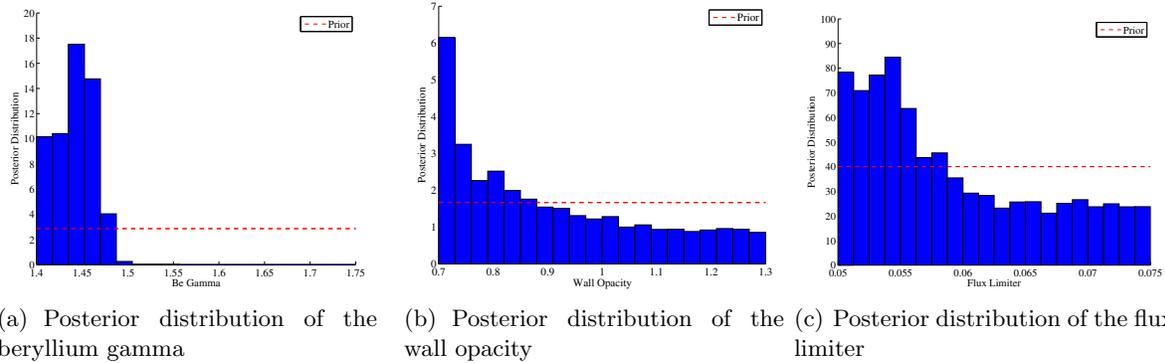
(a) Validation of the BMARS fit by leaving out then predicting 20 of the 104 H2D results.

(b) BMARS fit to all 104 H2D runs used for calibration.

**Figure 2: We evaluated the accuracy of the BMARS emulator by fitting to 84 of the 104 available runs and then predicting the remaining 20.**

Moving forward, we generated a latin hypercube sample of 50,000 points in the three-dimensional  $\vec{\theta}$  space. As we are only considering one diagnostic and its normally-distributed measurement error, our local likelihood calculations is exactly as written in Eq. (4). Figure 3 shows the posterior distributions of the three components of  $\vec{\theta}$  when they are calibrated to *all* of the asbo1 measurements.

The immediate observation is that all three components calibrated to the lower end of their



**Figure 3: The calibrated or posterior distributions of  $\vec{\theta}$  in the simplified case of considering only the asbo1 diagnostic.**

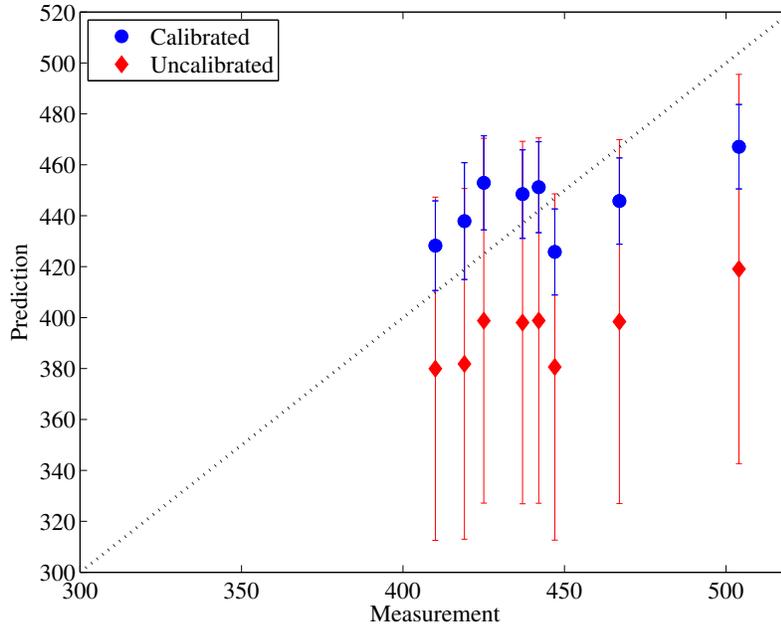
prior distributions and that the beryllium gamma input did so quite strongly. The interpretation of these plots is that the H2D simulations are more likely to replicate the experimental measurements when the uncertain inputs are sampled from the lower part of their uniform priors. It is difficult to provide physical meaning to the behavior of the calibrations; however, it is a bit surprising that the wall opacity calibrated at all, as its effect on the calculation of shock breakout time is mostly “downstream”. In other words, given the geometry of the problem, we do not expect that this input would affect the shock breakout time. This will be addressed in a following section.

The proof of concept is performed via leave-one-out type experiments: we calibrate on seven of the experiments and attempt to use the resulting posteriors to predict the 8th. We performed this calculation, in turn, for each of the eight asbo1 measurements. We will compare these predictions to predictions that would be made from blind sampling of the  $\vec{\theta}$  space. If the method is successful, we will improve the predictive accuracy and reduce the confidence interval about that prediction. Figure 4 compares predictions via sampling of the prior and calibrated  $\vec{\theta}$ s.

First we note that the predictive accuracy was greatly improved when we use calibrated  $\vec{\theta}$ s to generate predictions. The figure also indicates that the experimental measurements of shock breakout time fall into the upper-most range of the simulation responses. Recall that our model assumes that a discrepancy function (to account for systematic differences between the simulation and reality) is not necessary – in this case, we are very close to needing a discrepancy function to predict the asbo1 measurements. Indeed, one asbo1 measurements ( $\sim 500$ ps) is not encapsulated by the simulation runs. In the following section, we’ll show that the added information from the inclusion of all three diagnostics aides in the predictive accuracy and moves us away from nearly needing a discrepancy function.

### 3.2 Leave-one-out Experiments with Full Treatment of Uncertainty

We perform the same procedure of leave-one-out experiments and include treatment of the following uncertainties:



**Figure 4: The calibration routine improved the accuracy and reduced the uncertainty in experimental predictions.**

- Regression error - as before, we include the  $\zeta = N(0, \chi^2)$  error estimated by the BMARS routine.
- Beryllium disk thickness - micrometer measures to the nearest micron, so we include a  $U[\pm 0.5\mu\text{m}]$  error.
- Diagnostic measurement error - The measurement errors for the asbo1 (10ps), asbo2 (20ps), and sop (30ps) diagnostics are interpreted to be standard deviations. Since each is nominally a measurement of the same response, we take the average of the three to be the true experimental response and assume the worst case for measurement error,  $\epsilon \sim N(0, 30ps^2)$ .
- Systematic timing error - the laser firing mechanism has a systematic error of  $\pm 50ps$ . The lab's best interpretation of this error is that any breakout time within 50ps of the reported value is equally as likely as the reported value.

We treat the uncertainty in the disk thickness each time we sample a BMARS response by sampling a random number  $\pm 0.5$  microns from the nominal value we are interested in. For example, each time we compute a local weight, we are repeatedly evaluating BMARS responses at the nominal disk thickness and laser energy of one of the eight experiments. Instead of using the nominal disk thickness as an input, we use a randomly chosen thickness within a half-micron.

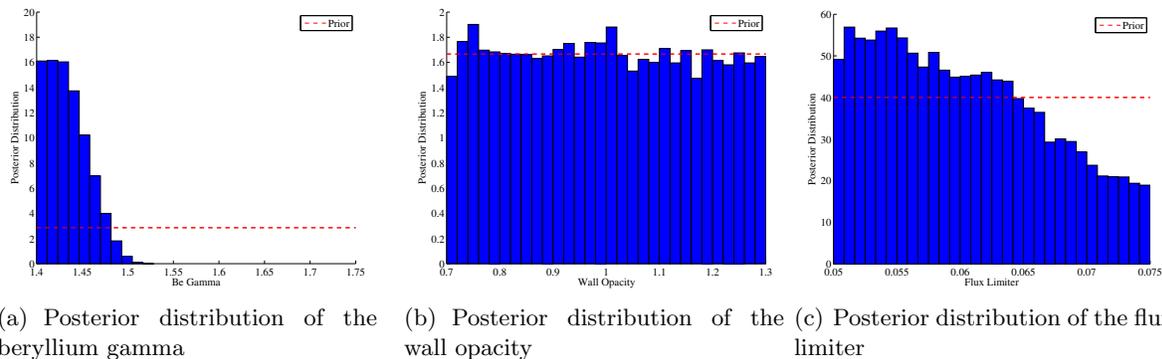
The data will show that using an average of the three diagnostics for calibration instead of a single diagnostic results in more consistent calibration. If we do use the average, however, we should retain some information about the spread of the three diagnostics. Considering the  $\pm 50ps$  box function about each measurement, we obtain a uniform range of possible true responses:  $U[\max_i - 50, \min_i + 50]$ , where  $\min_i$  and  $\max_i$  are the minimum and maximum measurements for

experiment  $i$ . Any  $\vec{\theta}$  whose average BMARS response falls outside this range is given a weight of zero (which will translate to a global weight of zero).

Finally, to further account for the spread in the experimental results, we add a third variance term to Eq. (4) which is equal to the variance of a uniform distribution with width equal to  $U[\max_i-50, \min_i+50]$ . The resulting posterior distributions from this calibration are given in Fig. 5.

First, it is clear that inclusion of all diagnostics, their uncertainty, and the uncertainty in the disk thickness resulted in a nearly complete loss of calibration in the wall opacity, and a less-sharp posterior distribution of the flux limiter. Both of these results are more intuitive: as explained above, the effect of the wall opacity is “downstream” from the shock breakout, so it should not be strong factor in the calibration of Hyades. We did expect some sensitivity to the flux limiter (a numerical parameter used to limit energy transport), and we were not surprised to see it calibrate to lower values because of our general believe that Hyades is over-driving the shock.

It is interesting that moving to the full treatment of uncertainty did not have a large effect on the posterior of the beryllium gamma constant. We did expect this to be a driving factor, since the shock is traveling through the beryllium, and we are modeling the ablated material with a  $\gamma$ -law equation of state. Thus, it seems intuitive that the value of the gamma constant would play a strong role in the calculation of shock breakout time. The fact that it remained strongly calibrated to the lower part of the prior indicates that the shock breakout time is highly sensitive to its value, and future H2D runs should probably explore more values below the bound of 1.4 explored in this data set.



**Figure 5: Including measurement, input, and systematic uncertainty in the calibration results in increased uncertainty in the wall opacity and flux limiter, but we retain almost the same posterior for beryllium gamma.**

The results of the leave-one-out predictions given these posteriors is illustrated in Fig. 6. Several differences between these predictions and those in Fig. 4 are noteworthy. First, as expected, the 90% predictive confidence intervals are larger, as they should be in response to the added sources of uncertainty. Second, we plot red horizontal error bars, which span the range of the  $\pm 50$ ps systematic uncertainty about the given measurements for each experiment. Three (or in one case, two) experimental responses are contained within each pair of these horizontal error bars. Finally, the horizontal coordinate of each prediction is simply the average of the experimental responses.

Despite the inclusion of multiple sources of uncertainty in this case, the root-mean-square predictive accuracy improved from 22.3ps to 16.2ps. As we alluded to above, we believe this is a result of using the mean of all three diagnostics as the target calibration value. Because the systematic timing error of  $\pm 50$ ps is dominating this study (this is more than 10% uncertainty in *either* direction!), the individual measurements of a single diagnostic are highly variable. The attempt to calibrate on only asbo1 in the previous section was complicated by this variability. In using an average of all three diagnostics, however, we were better able to characterize the true experimental response and the Hyades emulator was able to more consistently predict those values.

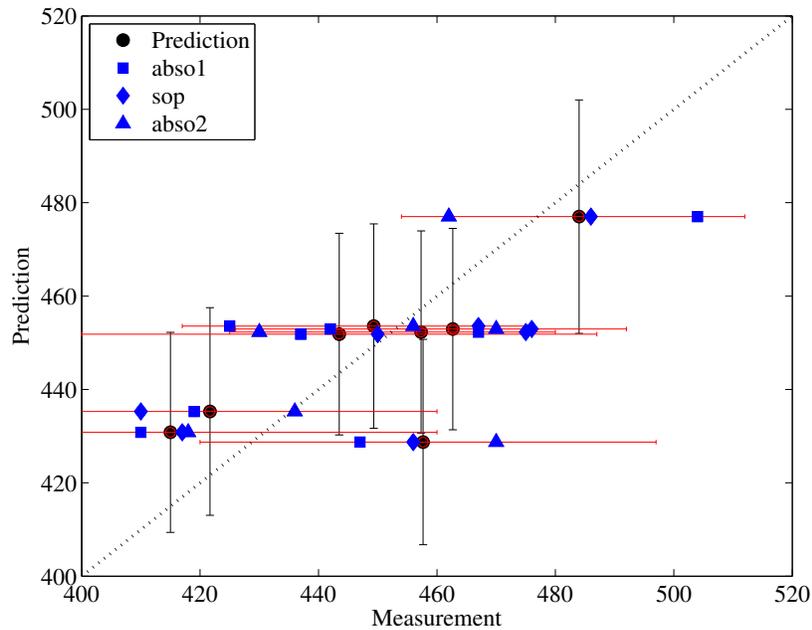


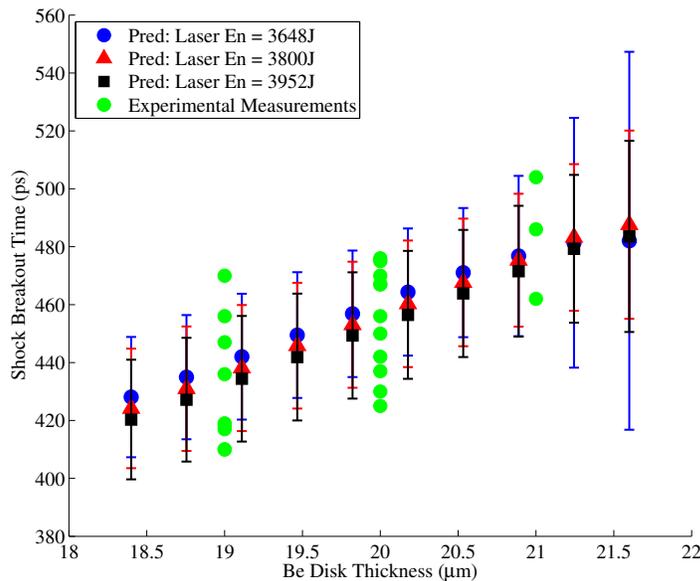
Figure 6: Leave-one-out predictions in the case of full treatment of uncertainty.

### 3.3 Predictions of New Shock Breakout Time Experiments

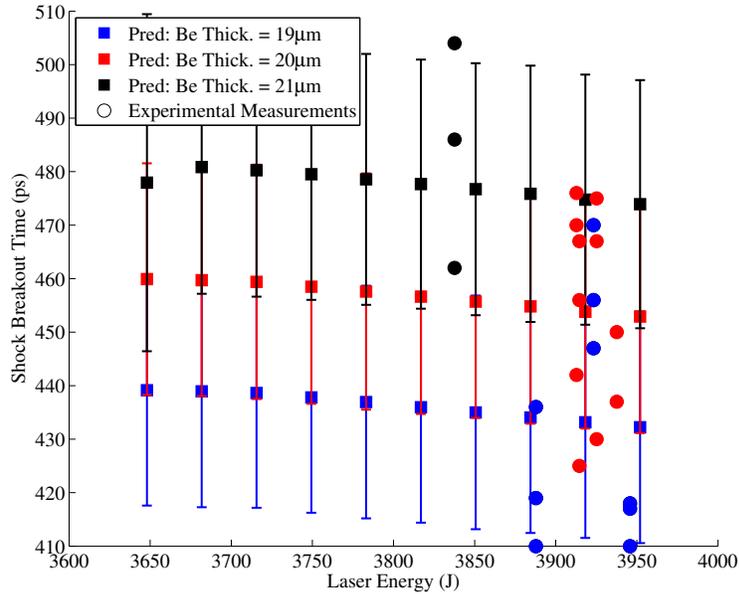
The prediction of the response of new experiments is the logical and practical application of any calibration method, and it is natural to extend the methods used to generate our leave-one-out predictions to predictions of the shock breakout time at new laser energies and/or disk thicknesses. The domain of valid predictions is, at most, the range of the laser energy and

disk thickness samples from the original H2D data. Prediction outside this range would be an extrapolation of the emulator. We also must note that extrapolation outside the range of the experimental data requires an assumption that  $\vec{\theta}$  would calibrate similarly at any given  $\vec{x}$  and that H2D remains a valid model outside the range of the experimental data. The degree to which this is true will typically require expert judgment.

Predictions of breakout time for new experiments is complicated by an additional factor in this application. The laser energy of a given experiment is accurately known after the experiment; before the experiment, however, the facility estimates a laser energy within  $N(0, 19.4J^2)$  of the requested amount. Therefore, as with the disk thicknesses, each time we evaluate a BMARS model, we randomly sample a number within this distribution of the nominal experimental value. This should add considerable width to our confidence bounds of new predictions. Figures 7 and 8 show predictions of shock breakout time as a function of disk thickness and laser energy, respectively. Note that each figure shows results from holding the other independent input constant at three values.



**Figure 7: Predictions of breakout time at new disk thicknesses for three different laser energies,**



**Figure 8: Predictions of breakout time at new laser energies for three different disk thicknesses. In this figure, the color of the experimental data points corresponds to the disk thickness for that experiment, as indicated by the legend.**

The figures indicate that in the range of inputs tested here, breakout time is a stronger function of disk thickness than of laser energy. The predictions are consistent with the experimental data and intuition in that breakout time increases with increasing disk thickness and decreasing laser energy. Our predictions mostly contain the experimental data points (recall that we calibrated on the mean of three measurements per experiment, so our predictions really trace the mean of each experiment). At new experiments, we see that our predictions are fairly smooth and extrapolate the general behavior of the experimental data points. In Fig. 7 we see some evidence (increasing confidence bounds) of extrapolation of the emulator at the larger disk thicknesses. This should tell the modeler that predictions in this domain are less certain.

### 3.4 Discussion of the Method's Performance and Possible Limitations

As we pointed out in the introduction, the method we presented is analogous to an MCMC algorithm with uniform sampling, except that we do not consider the state of the Markov chain when evaluating the samples. Also, instead of simply accepting or rejecting the samples, we apply a weight that should be representative of the likelihood of that sample. In our specific application, we found cause to assign a weight of zero to some samples, which is equivalent to rejecting the sample. Thus, we claim that the method is an extension of Metropolis-Hastings[4][1] algorithms in that it includes a measure of importance of the samples.

There are a number of areas for exploration regarding this method. For example, we only considered uniform sampling of the  $\theta$  parameter space under the assumption that we have no

prior information about our uncertain inputs. If some prior information can be estimated, then the modeler may consider alternate sampling strategies or a modification of the computed weights to reflect the likelihood of the sample given the prior distribution(s). Also, the application presented here is relatively low dimensional ( $p=3$ ). This allowed extremely dense sampling of the  $\vec{\theta}$  space and a very large number of BMARS evaluations at each sample. The computational cost would obviously increase with the dimensionality of the  $\vec{\theta}$  space, and it is likely that the sampling efficiency would become inadequate for some higher-dimensional problems.

#### 4. CONCLUSIONS

We have outlined a method for calibrating the uncertain inputs to a computer model using experimentally measured data. The method requires sampling of the uncertain parameter space, and each of these samples is assigned a weight based on the simulator response at that sample and the uncertainties involved in the problem of interest. Most practical applications will require an emulator to generate samples of the simulator response. Typical sources of uncertainty will include emulator regression error, uncertainties in experimental inputs, and experimental measurement error. The weights that are generated by the algorithm are used to generate posterior distributions and weighted predictions of new experiments, including confidence intervals.

We applied the method to a calibration of the Hyades 2D laser deposition model using experimentally measured shock breakout times. We employed the use of the BMARS emulator and tailored our weighting computation to include the realistic uncertainties involved with our experimental data. The results of leave-one-out type predictions indicated that the method greatly improved predictive accuracy and tightened the confidence intervals about those predictions. With the successful leave-one-out tests as validation, we extended the method to produce predictions (and confidence bounds) for new experiments.

#### ACKNOWLEDGEMENTS

The first author also wishes to acknowledge a number of members of the research staff at Lawrence Livermore National Laboratory for helpful conversations regarding model calibration: P. Dykema, S. Brandon, G. Johannesson, and B. Johnson. Funding for the first author was provided by the Department of Energy Computational Science Fellowship under grant number DE-FG02-97ER25308. The work of the other authors was supported by the DOE NNSA/ASC under the Predictive Science Academic Alliance Program by grant number DEFC52-08NA28616.

#### REFERENCES

1. L. Tierney. Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics.*, **Vol. 22, No. 4**, pp. 1701-1728 (1994).
2. D.G.T. Denison, B.K. Mallick, and A.F.M. Smith. Bayesian MARS *Statistics and Computing.*, **Vol. 8, No. 4**, pp. 337-346 (1998).
3. J.H. Friedman. Multivariate Adaptive Regression Splines. *The Annals of Statistics.*, **Vol. 19, No. 1**, pp. 1-67 (1991).

4. S. Chib and E. Greenberg. Understanding the Metropolis-Hastings Algorithm *The American Statistician.*, **Vol. 49, No. 4**, pp. 327-335 (1995).