Reliability Engineering and System Safety 96 (2011) 1194–1207

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

A physics informed emulator for laser-driven radiating shock simulations

Ryan G. McClarren [a,*], D. Ryu [b], R. Paul Drake [c], Michael Grosskopf [c], Derek Bingham [f], Chuan-Chih Chou [c], Bruce Fryxell [c], Bart van der Holst [c], James Paul Holloway [d], Carolyn C. Kuranz [c], Bani Mallick [b], Erica Rutter [c], Ben R. Torralva [e]

[a] Department of Nuclear Engineering, Texas A&M University, College Station, TX 77843-3133, United States
[b] Department of Statistics, Texas A&M University, College Station, TX 77843-3143, United States
[c] Atmospheric Oceanic and Space Sciences, Space Physics Research Laboratory, University of Michigan, Ann Arbor, MI 48109, United States
[d] Department of Nuclear Engineering and Radiological Sciences, University of Michigan, Ann Arbor, MI 48109, United States
[e] Department of Materials Science and Engineering, University of Michigan, Ann Arbor, MI 48109, United States
[f] Department of Statistics and Actuarial Science, Simon Fraser University, Burnaby, BC, Canada

ARTICLE INFO

Article history:
Received 18 March 2010
Received in revised form
3 August 2010
Accepted 4 August 2010
Available online 16 April 2011

Keywords:
Uncertainty Quantification
Data Reduction
Bayesian MARS
Gaussian Process Regression

ABSTRACT

This work discusses the uncertainty quantification aspect of quantification of margin and uncertainty (QMU) in the context of two linked computer codes. Specifically, we present a physics based reduction technique to deal with functional data from the first code and then develop an emulator for this reduced data. Our particular application deals with conditions created by laser deposition in a radiating shock experiment modeled using the Lagrangian, radiation-hydrodynamics code Hyades. Our goal is to construct an emulator and perform a sensitivity analysis of the functional output from Hyades to be used as an initial condition for a three-dimensional code that will compute the evolution of the radiating shock at later times. Initial attempts at purely statistical data reduction techniques, were not successful at reducing the number of parameters required to describe the Hyades output. We decided on an alternate approach using physical arguments to decide what features/locations of the output were relevant (e.g., the location of the shock front or the location of the maximum pressure) and then used a piecewise linear fit between these locations. This reduced the number of outputs needed from the emulator to 40, down from the O(1000) points in the Hyades output. Then, using Bayesian MARS and Gaussian process regression, we were able to build emulators for Hyades and study sensitivities to input parameters.

1. Introduction

The quantification of margins and uncertainty (QMU) is a topic that has received considerable attention of late, particularly by the National Nuclear Security Agency, as reflected for example in the recent National Research Council study [1]. A core component of QMU is uncertainty quantification, and, as emphasized by the NRC report, there is a strong need to further develop the methodology for this work, especially in using computational models to predict, with uncertainty, the threshold or design values for successful operation of an engineered system. Another aspect stressed in the report is that there should not be a monolithic QMU approach: a diversity of approaches to determining the margins and quantifying uncertainty needs to be actively developed.

In performing QMU analyses on complex, multiphysics systems the task can be further complicated by the fact that several computer codes may be linked together (i.e., the output from one code becomes the input for another). In such a case the input parameters for a computer code are a field of values rather than several discrete inputs. Therefore, to robustly predict the behavior of the system will require characterizing the uncertainty/sensitivity of the system to changes in any one of the numbers that comprise these field inputs.

As part of a related problem, the task of QMU is especially challenging when the number of uncertain input parameters or output quantities of interest is large. In such a case exploring the parameter space to ensure acceptable operation of the system may not be achievable via sampling techniques especially if performing computer simulations is expensive. Moreover, for robustness purposes perturbation and first-order reliability methods may not be adequate. In such a case it may be desirable to construct an emulator for the computer code.

An emulator attempts to reduce the computer code output into a regression model. If one thinks of a computer code as a, complicated perhaps, function $f(\vec{x})$ that maps input values to some number of output quantities, the emulator is a fit $\hat{f}(\vec{x}) \approx f(\vec{x})$. To construct an

* Corresponding author.
E-mail address: rgm@tamu.edu (R.G. McClarren).

0951-8320/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ress.2010.08.012

emulator one does a series of computer simulations by varying the input values and then uses the outputs of those simulations to build a regression model. Emulators are obviously useful in the case where the computer code is expensive to run because evaluations of $\hat{f}$ are basically free. Of course in the QMU enterprise one cannot rely on an emulator to draw conclusions about a system behavior because the regression model might miss important aspects of the simulation due to its approximate nature. Nevertheless, one can use an emulator to define regions of the input space where the system will not perform acceptably and then use additional computer simulations to confirm the performance. Also, for calibration of a computer code [2] or building a predictive model [3,4] as a part of QMU the emulator can be used to rapidly generate samples of the computer code.

In this paper we deal with a situation where one desires to construct an emulator for a computer code that is part of a chain of linked computer codes. This requires constructing an emulator that can reproduce field data. To accomplish this task we first develop a method to reduce the field data to O(10) degrees of freedom. This data reduction is accomplished by using physics-based reasoning rather than statistical techniques. These degrees of freedom can be used to reproduce, with acceptable error, the field data. Then we construct an emulator to predict the values of these output degrees of freedom given a set of inputs to the computer code. In this manner we can generate the field data that is the input to the second code using only the emulator. We believe that our approach could have wide applicability to QMU practitioners or others performing uncertainty analyses in the situation where several linked computer codes are a part of the simulation strategy. Though we have not explored this potential application, we surmise this data reduction strategy could also be useful in linking codes solving different scales of a physical problem, such as linking atomistic to mesoscale to engineering scale codes. Our method contrasts with previously described, generic approaches for dealing with functional data based on principal component analysis [2] and wavelet decompositions [5]. In the absence of a physics-based data reduction strategy these methods would be an excellent starting place. Moreover, we are optimistic that some hybrid of our approaches would likely be a fruitful avenue for future research.

The motivation for this work was the uncertainty quantification of simulations relevant to the mission of the Center for Radiating Shock Hydrodynamics (CRASH), funded by the Department of Energy Predictive Science Academic Alliance Program (PSAAP). At CRASH we endeavor to simulate and predict the behavior of laser-driven shock waves traveling at high Mach number down a gas-filled tube. The temperatures reached in the experiment are high enough that energy carried by X-ray radiation affects the dynamics of the shock evolution. To simulate these shocks an Eulerian radiation hydrodynamics code, the CRASH code, has been developed based on a 3D, adaptive, massively parallel magnetohydrodynamics code [6]. Using the CRASH code, we will predict experimentally observable quantities such as shock location as a function of time. To initialize the CRASH code we use a Lagrangian radiation-hydrodynamics code, Hyades [7], that computes the laser energy deposition and early time shock formation. Therefore, we take the field data for the hydrodynamics and radiation variables from Hyades (approaching 10^3 outputs even for 1D simulations with simplified physics) to initialize the CRASH simulation. As a result, to compute the sensitivity/uncertainty of the CRASH predictions for the result of an experiment, we need to know the sensitivity of the Hyades outputs to its uncertain inputs. If we had to compute sensitivities to each of the Hyades outputs, this task would be nearly hopeless.

We have, however, been able to reduce the number of outputs required to adequately characterize the Hyades results using physical insight. The CRASH output is not sensitive to every detail of the Hyades output, and we have leveraged this fact to characterize the Hyades output using only 40 parameters. We can compute the sensitivity of these 40 parameters to quantify how much the uncertainties in the input to Hyades affect its output. As a result we have identified the areas of input space where we need to focus effort in reducing uncertainties. This data reduction has the added benefit of making the linking of the two codes more automated and reproducible. Finally, as part of our strategy for using Hyades to initialize CRASH we have performed a series of experiments to be used to calibrate the Hyades output. By reducing the field data and creating an emulator, this future calibration exercise can be performed much more rapidly.

The overarching goal of CRASH is to use experiments and simulation studies from the first four years of the project to build the capability to model the so-called year five experiment that is an extrapolation beyond the previously performed experiments. As shown in Fig. 1, the base experiment has a laser-driven shock launched down a cylindrical tube of xenon. The year-4 experiment complicates the geometry by having the shock initially launched into a larger cylindrical tube that necks down into a smaller tube. These experiments, and their accompanying simulation studies, will be used to inform our ability to predict several quantities of interest (e.g., shock position) in the year-5 experiment where a cylindrical tube necks down into a elliptical tube. To be able to quantify the uncertainties in our prediction for the year-5 experiment we will need to be able to justify that:

- Our physics models are adequate (i.e., that we can validate our models for this particular experiment).
- Our code is behaving in the way we expect (i.e., our code has been through several relevant verification exercises).
- We understand how uncertainties in the inputs affect the output of the code and our prediction.

All of these justifications are being compiled into an exhaustive evidence file. Indeed we studied to what extent our models are

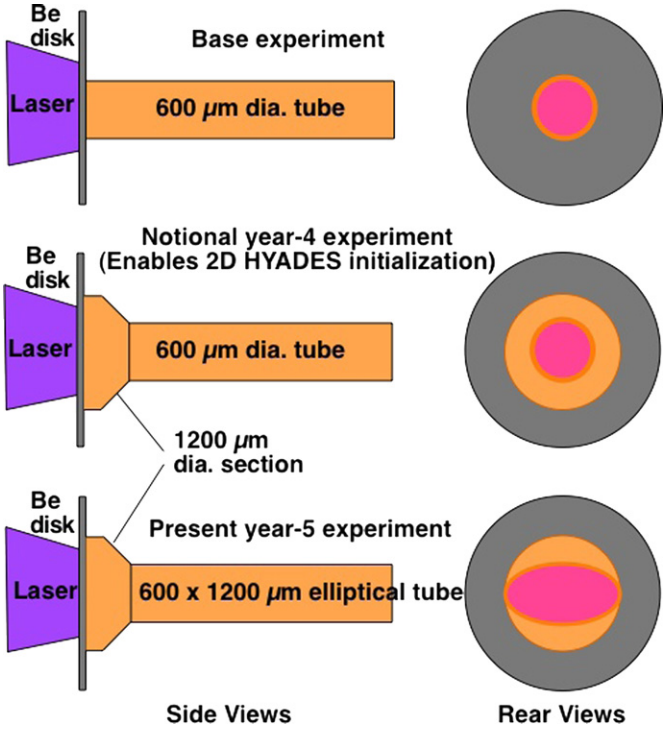

Fig. 1. Diagrams of the CRASH experiments from the base experiment to the year-5 extrapolation experiment. The shock tubes are filled with xenon gas.

"valid" for the simulations of interest. A discussion of how this validation is coupled to uncertainty quantification can be found in [8].

The activities at CRASH do deal with risk and margin, but perhaps in ways that may not be obvious. Given that we must predict the several quantities of interest on an experiment in a geometry on which we have no data, our primary risk is that our computational strategy will not be sufficient to predict with quantifiable uncertainty the results of the year five experiment. To deal with this risk we have delineated several strategies to add higher fidelity physics models, if necessary. We are performing detailed uncertainty quantification and predictive modeling on the experimental data that we do have to assess this risk. For a description of such a predictive modeling exercise see a companion paper in this issue by Holloway, et al. [8]. The concept of margin for our predictive science campaign is more nebulous. All expert judgment indicates that the year five experiment will "work" insofar as a radiative shock will launched down the tube. Our margin is in the computer simulations: how wrong can our physical models be without making our predictions fail? The work outlined in this paper seeks to answer questions about both the risk and the margin in part of the computational strategy at CRASH. To assess the sensitivity of the output of Hyades and its impact on the CRASH simulation we needed to both reduce the field data to a few degrees of freedom, and build an emulator to explore the range of Hyades inputs. The results pointed to us input uncertainties that were most affecting our computations. Therefore, to mitigate the risk associated with these uncertainties we must take steps to minimize them.

The outline of this paper is as follows. The next section describes the basics of the radiating shock experiments and the simulations we perform. This is followed in Section 3 by a discussion of how the data from Hyades are partitioned using physical motivation. In Section 4 we discuss the two approaches we used to develop an emulator for the partitioned Hyades data, and Section 5 describes the actual Hyades run set used to build the emulator. Then we describe the results of the emulator construction and perform a sensitivity analysis in Section 6. Section 7 details how we used the emulator output in a CRASH uncertainty quantification (UQ) study, and we conclude the paper in Section 8.

2. Description of the experiment and corresponding simulation

Here we give an overview of our experiments and simulation; for the details of the underlying physics of the radiative shocks see, for example, [9–11]. In these experiments a laser pulse irradiates a thin disk of beryllium (Be) metal at the end of a xenon (Xe) filled tube (see Fig. 1 for a diagram). The energy of the laser causes the surface of the Be to ablate. To balance the momentum of the ablating material, a shock wave is driven into the Be at about 50 km/s. When this shock wave breaks out of the Be into the Xe, it is propagating at a speed of roughly 200 km/s and heats the Xe to temperatures above 5×10^5 K. At these high temperatures the Xe becomes a plasma and emits a great deal of energy in the form of soft X-ray radiation. This radiation travels ahead of the shock, heating the unshocked xenon and providing feedback to the shock dynamics. Such a situation where radiation affects the evolution of a material system is a topic of applied physics known as radiation hydrodynamics. A salient difference between radiation hydrodynamics and ordinary hydrodynamics is that in ordinary hydrodynamics material ahead of a shock does not know the shock is coming because the shock travels faster than the speed of sound in the material. The presence of radiation energy upsets this classical picture of a shock and introduces new phenomena. The particular radiating shock experiments with which we are concerned can be viewed as scaled experiments for understanding astrophysical shock waves and other high temperature phenomena [11].

Using computer simulation we seek to predict several features of the shock, such as its position down the tube as a function of time and the thickness of the layer of shocked Xe. The paradigm we have adopted uses a Lagrangian radiation hydrodynamics code, Hyades, to compute the laser energy deposition and system evolution for the first 1.3 ns (the laser pulse duration is 1 ns at full width half maximum). The result of the Hyades computation is an initial condition for the CRASH code, an adaptive mesh refinement (AMR) Eulerian radiation hydrodynamics code that computes the shock behavior in the Xe. This hand-off from Hyades to CRASH is necessitated by the mesh tangling that results when modeling complex flows using a multidimensional Lagrangian code.[1] This mesh-tangling problem is so severe that Hyades cannot evolve the system up to times at which the experimental observations are made. CRASH cannot, however, model the laser absorption; therefore, modeling at early times with Hyades is necessary unless one were to add a laser package to the CRASH software. The decision to try and use Hyades coupled to CRASH does introduce some risk in terms of having an accurate simulation. Namely, by coupling two codes there is some risk than an error in the coupling will decrease our margins to acceptably model the experiment. There is, however, a risk mitigation plan in place. If the linked arrangement is not able to model the data from the base experiment and we can reasonably infer that the linking is the culprit, we can choose to invest resources into adding a laser package into CRASH. If, on the other hand, the Hyades-CRASH link can acceptably simulate the results from the base experiment, we should be able to justify that this positive result will carry over to the year-5 experiment.

To develop confidence in this linked arrangement between codes, we need to understand the sensitivity of the Hyades results as a function of (1) experimental conditions such as laser irradiance, Be disk thickness, Xe gas pressure, etc. (2) numerical parameters such as the number of mesh points or the number of energy groups. Moreover, because we are interested in using the output from Hyades to initialize the main simulation code, we are presented with a large number of parameters (hundreds of mesh points times the number of hydrodynamic variables) for which we desire sensitivity information. In the next section we discuss our approach for reducing the number of parameters needed to characterize the Hyades output. Later we will describe the process of generating a regression model for the Hyades output.

3. Physics-informed partitioning of Hyades output

One might hope to develop an analytic model of the structure of the Hyades output at a given time. For our application this would be the time at which the Hyades output is used to initialize the CRASH code. This would offer the merit of having physically based parameters that could be adjusted to fit the Hyades results. Unfortunately, this turns out to be quite complex because the effects of the laser-heated electrons continue for several hundred picoseconds after the end of the laser irradiation. Alternatively, one might hope to do fitting of the Hyades output to identify a

[1] Mesh tangling happens because a Lagrangian hydrodynamics code moves the spatial mesh along with the flow of material. When there is significant shearing or other mixing, the mesh nodes can cross ("tangle") creating negative volume cells [12]. This tangling of the mesh is often unavoidable in multi-dimensional flows and usually results in the simulation crashing.

simple, parameterized, description of the structure. Our initial attempts to do this by purely statistical methods, such as the Bayesian Partition model [13], did not go well, in part because this is a purely statistical model that does not benefit from the physical principles we take for granted. In particular, the statistical methods indicated that we needed 150 degrees of freedom to describe the Hyades data. For example, this, or indeed any, statistical model cannot know that temperature must always be positive without explicitly adding this to the model. To cope with this shortcoming we approached the data reduction from a different perspective: we developed a physically motivated partitioning of the Hyades 1D output. This made the development of an emulator much more efficient because we needed to predict the simulation output of many fewer degrees of freedom.

Fig. 2 shows the velocity, density, and pressure profiles from a 1D Hyades run at a time of 1.25 ns (the initialization is standardized at 1.3 ns and with a position of zero corresponding to the left edge of the Be disk). The link time between Hyades in CRASH is constant across our linked runs so as to not introduce additional uncertainty in the initialization of CRASH. This run is tuned to match the observed location of the shock in the Xe gas at 13 ns [14]. The smooth curves show the Hyades profiles while the solid, piecewise-linear curves show a interpolated fit. First we discuss the origin of the observed structures. The laser irradiates the Be disk, first driving a shock wave through it. The shock wave breaks out of the rear of the Be disk in about 500 ps, after which two things happen. First, the rarefaction ahead of the rear surface drives a shock into the Xe gas at just over 100 km/s. Second, a pressure gradient develops from the released rear material toward the denser material heated by electron heat transport from the laser absorption region. This pressure gradient accelerates the bulk of the Be, a process that can be simply modeled as rocket acceleration. By the decline of the laser irradiance from 1 to 1.1 ns, the bulk of the Be has reached the same velocity as the initial shock, about 120 km/s.

Nevertheless, the end of the laser irradiation is not the end of the Be acceleration, because the pressure gradient that accelerates the Be remains present until the electrons cool by expansion and heat conduction. As the electrons cool, the pressure profile develops the peaked structure seen in Fig. 2. The Be to the right of the peak continues to be accelerated, while the Be to the left of the peak is decelerated. As a result, the region to the left of the peak has very little impact on the subsequent dynamics. One would expect that there is no need to model this precisely, or even to include the material far enough from the peak, in order to accurately initiate the CRASH calculation. The pressure drops with time, so that by 2 ns the pressure profile has flattened out and no longer accelerates the Be. Much later on, the structure evolves toward that of a blast wave in which the pressure accelerates material gradually away from the shock.

In the figure, one can see a region of reduced pressure gradient to the right of the peak. This is a remnant of the initial launching of the shock in the Xe at shock breakout. If one follows the line upward from the left boundary (near $x=0$) of this flat region, one can see that this corresponds to the location of maximum velocity. The further acceleration of the Be after 1 ns has launched a velocity impulse forward through the leading edge of the Be. The maximum velocity is about 180 km/s at 1.25 ns and occurs at $x=0.0125$ cm, and in this calculation reaches 220 km/s by 1.5 ns when the velocity impulse has overtaken the shock in the Xe. This corresponds to the maximum post-shock ion temperature found by Hyades, which is about 2 keV. At 1.25 ns, the shock is established in the Xe but is most evident in the figure in the density. At this time, the immediate post-shock ion temperature is about 700 eV. The structure in the Xe is not well resolved at this time, with both the pressure and the velocity showing gradual transitions. This is unavoidable in the context of doing a viable 1D Lagrangian model.

The density maximum is located between the maxima of pressure and velocity, and is the natural result of the ablation of Be to the left and the expansion of the rear Be surface to the right. Once the velocity maximum has overtaken the shock, the velocity profile becomes and stays quite linear. This is typical of freely evolving hydrodynamic systems and corresponds to steady expansion with time. As the system expands, the density decreases and the density maximum eventually disappears.

With the above context, the following assumptions seem reasonable for construction of a fit to the Hyades output: (1) Material having significant negative velocity can be approximated by very simple and inaccurate profiles, because it just continues to slowly accelerate to the left and cannot impact the dynamics of interest. The fit shown in the figure and described here ignores the laser-heated corona at low density and approximately captures the exponential density profile from the material that was heated by electron heat transport. (2) The initial state of the radiation precursor does not need to be modeled in detail, because the energy through the shock by 1.3 ns is less than 10% of the energy by 13 ns. We demonstrated this in test runs with CRASH that removed the initial precursor heating.

First, the minimum position of the Hyades output is needed. The fit shown uses half this value as its limit, because, as one can see, the density and pressure profiles fall off much more steeply (due to ignoring the laser heated corona). Call this value $x_{\min}$. Second, one needs the position, velocity, density, and pressure at the locations where (from left to right):

- the velocity first exceeds -3×10^7;
- the velocity is $\frac{1}{2}$ the maximum value;
- the derivative of the pressure abruptly decreases;
- the derivative of the pressure abruptly decreases again (becoming negative or more negative);
- the density is maximum;
- the velocity is maximum;
- the interface is, from the Be side;
- the interface is, from the Xe side;
- the shock is located;
- the precursor properties are steady.

The data just described need to be ordered so that the position monotonically increases. The fit then is piecewise linear in all regions, beginning at $x_{\min}$, except for the density and pressure, which are fit as linear exponentials between the first two locations from this list and left from there to $x_{\min}$. There is no need to fit the details of the shock structure, both because they are unimportant for the long-term dynamics and because they are incorrect as represented by Hyades. The piecewise linear fit shown in Fig. 2 used the list of values shown in Table 1. One could calibrate the fit to preserve some defined quantity of mass, momentum, or energy, but at the present level of detail this would be overkill. As noted above we allow a large discrepancy between the fit and simulation output to the left of the Be disk edge because the values at these locations do not influence the behavior of what happens in the Xe tube: the material is outside the tube and moving to the left.

It is hard to see how fewer locations might adequately represent the physical system, except that one might drop the one with the most negative velocity. The upshot here is that the profiles are minimally represented by four parameters at 10 locations. The total of 40 parameters include some that might be inferred from correlations, but not many. We refer to this partitioning, especially when in the context of an emulator, as the physics informed emulator (PIE).

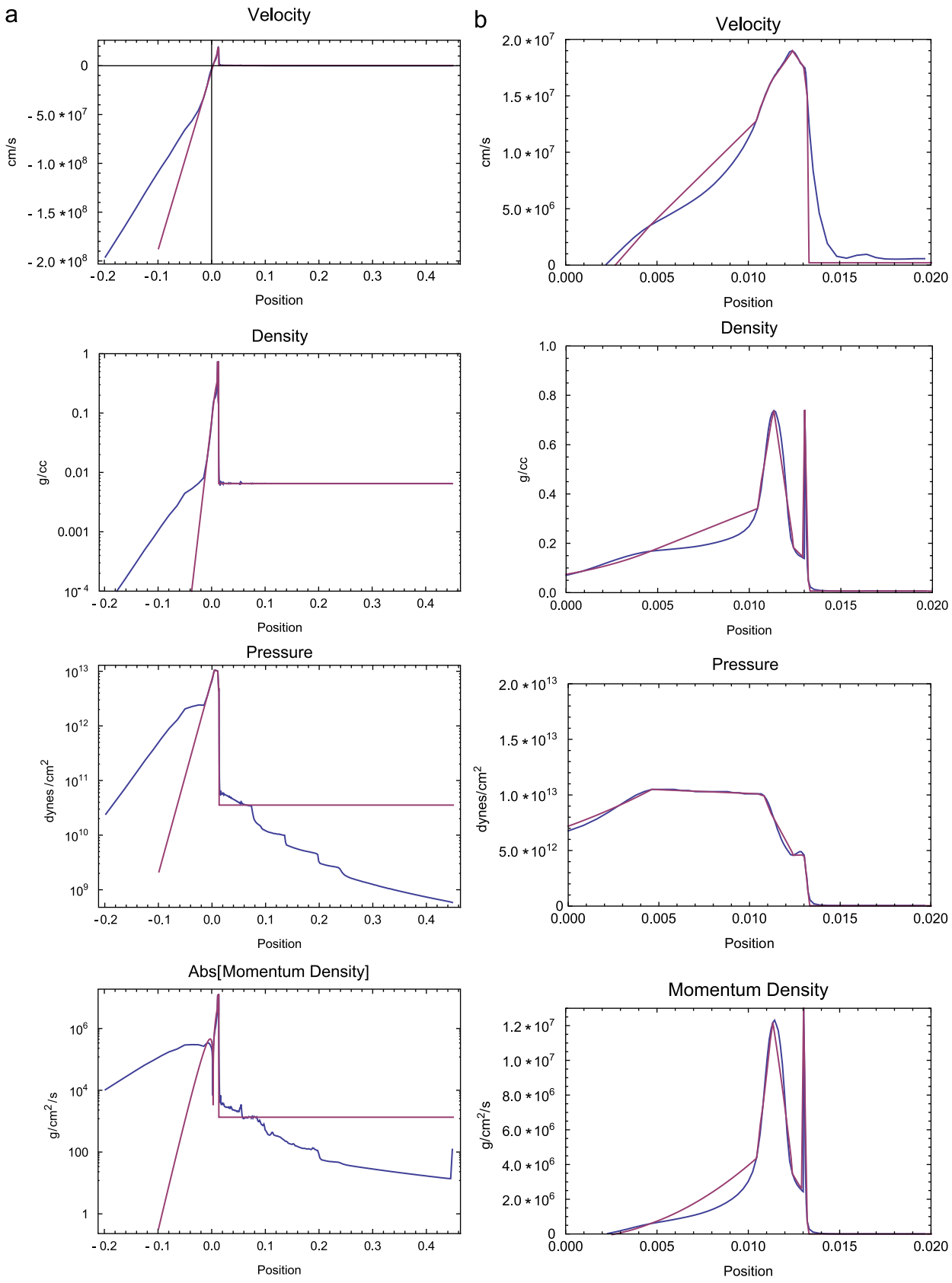

Fig. 2. Hyades output (blue) at 1.25 ns compared with piecewise linear fit from the physics informed emulator (red). The position of zero corresponds to the left edge of the Be disk: (a) Hyades results and (b) detail of (a). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Parameters from 1D Hyades (cgs units).

	Position	Velocity	Density	Pressure	Material
$u > -3 \times 10^7$	−0.008363001	−2.04E+07	0.016829163	3.62E+12	Be
p left corner	0.004615684	3.50E+06	0.168068666	1.05E+13	Be
ρ half max	0.010460472	1.29E+07	0.342857445	1.01E+13	Be
p rt corner	0.010801793	1.45E+07	0.510666015	9.92E+12	Be
ρ max	0.011342457	1.65E+07	0.738250317	7.88E+12	Be
u max	0.012412547	1.90E+07	0.182946	4.57E+12	Be
Be at interface	0.013004106	1.76E+07	0.138409625	4.61E+12	Be
Xe at interface	0.01302491	1.76E+07	0.740472601	4.43E+12	Xe
Shock	0.013222783	1.50E+07	0.049271551	1.35E+12	Xe
Precursor	0.063511755	2.07E+05	0.006476877	3.55E+10	Xe

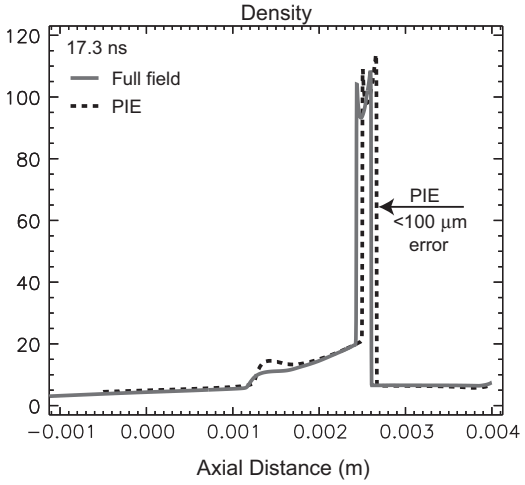

Fig. 3. Comparison of density at 17.3 ns from CRASH simulations initialized with the full Hyades output to simulations initialized using the physics informed emulator (PIE).

To verify that these 40 parameters are adequate to describe the Hyades data used to initialize CRASH, we compared simulations of the shock evolution using CRASH with the nominal configuration (see Section 5) using the full Hyades output to simulations initialized using the 40 parameters of the physics motivated partitioning of the data. In Fig. 3 we show a sample of the results of these simulations. In this particular example we see that the different initializations do affect the simulation output somewhat. Nevertheless, the change in shock position is on the order of our experimental uncertainty, which is on the order of 100 mm due to several factors including timing of the radiography and projecting the shock tube onto planar film. Other features in the solution have changed slightly: behind the shock, the velocity, pressure, and density have a slightly different shape between 0.001 and 0.002 m. Roughly speaking one could consider this a rudimentary validation of the process of using the 40 parameters to replace the Hyades field data at 1.3 ns for this particular experimental set up. Without a more formal analysis of how accurate this fit is, we cannot rely on it in our final predictions for the year-5 experiment. We can, however, use this fit for nearby experiments and to develop predictive models based on experimental data.

4. Emulators

Given a reduction of the Hyades field data to a small number of degrees of freedom, we are in a position to construct an emulator for the Hyades code for these experimental configurations. Before detailing how this is done we will first outline the reasoning for constructing an emulator. Given that Hyades is proprietary software that can only be run in serial calculations, we want to be able to calibrate its output, as mentioned before, and determine its sensitivities and uncertainties over given ranges of input parameters. Given the number of input parameters and the fact that high resolution calculations with this serial code are expensive, we desire a means to generate efficient, approximate results from Hyades calculations for calibration and uncertainty quantification purposes. This was the primary motivation for building an emulator, and to this end we did hundreds of Hyades calculations varying the relevant input parameters. Extracting the 40 PIE parameters from these runs we were able to build a regression model that generates a response surface for each of the 40 degrees of freedom.

We can also use an emulator to generate rapid variations in input conditions to the CRASH code. This would be useful for generating on the fly simulations between laser shots during an experimental campaign (the CRASH experiments generally consist of about 10 instantiations of the experiment over the course of one day at the Omega laser). One feature of the laser facility we use is that the actual laser energy put into the system is known after the shot; we can request a particular laser energy but due to variability in the laser system the actual amount of energy delivered will vary. With an emulator we can rapidly generate information about the experiment between shots to possibly adjust diagnostics and improve the usefulness of later shots.

It should be noted that for high consequence simulations one must be skeptical of using an emulator. The errors introduced by approximating a simulation run might lead to wholly incorrect predictions of system performance. Rather we believe that an emulator is most useful in constructing predictive models and exploring the uncertainties and sensitivity of a code and generating a rough idea of the failure points of the system. Using an emulator in such a way, can greatly reduce the number of heroic computer simulations needed to assess the performance of the system.

To construct an emulator, we consider a regression model for the responses of the outputs Y_i, $i=1,\ldots,n$, from n runs of simulations or experiments on the predictors or the inputs $\boldsymbol{X}_i=(X_{i1}, \ldots, X_{im})$ such that

$$Y_i = f(\boldsymbol{X}_i) + \varepsilon_i \quad \text{and} \quad \boldsymbol{X}_i \in D \subset \mathbb{R}^m, \tag{4.1}$$

where f is an unknown regression function that we wish to estimate, ε_i is a random error with zero mean, usually assumed to be from a Gaussian distribution, and D is the domain of interest, e.g., convex hull defined by the predictors.

In our case, because we are dealing with computer simulation and not performing a measurement, the value of ε is zero. Each of the 40 PIE parameters for a given run of Hyades is a Y_i that we will model independent of the other 39 parameters. Modeling these

parameters as related is the topic of ongoing research, including the use of seemingly unrelated regression models [15]. In principle we should be able make use of the known relations between the parameters. For instance, the parameters are ordered so that the position is an increasing function, and the other values of density are all less than the density maximum.

The two approaches used to construct a regression model are detailed in the remainder of this section.

4.1. Multivariate adaptive regression splines (MARS)

Multivariate adaptive regression splines (MARS) [16] is a nonparametric regression. With k basis functions B_i and suitable coefficients a_i, $i=1,\ldots,k$, the MARS estimates f such that

$$\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{k} a_i B_i(\boldsymbol{x}), \tag{4.2}$$

where $\boldsymbol{x} \in D$. The basis function B_i with the degree of the interaction J_i consists of the sign indicators $s_{ij} = \pm 1$, knot points t_{ij}, and the order r_i

$$B_i(\boldsymbol{x}) = \begin{cases} 1, & i = 1, \\ \prod_{j=1}^{J_i} [s_{ij}(x_{v(i,j)} - t_{ij})^{r_i}]_+, & i = 2,3,\ldots, \end{cases} \tag{4.3}$$

where $(\cdot)_+ = \max(0,\cdot)$, and $v(i,j)$ gives the index of the predictor variable split on t_{ij}. The optimum basis functions including knot points can be achieved by the generalized cross-validation criterion [17]. Bayesian MARS (BMARS) [18] assigns a prior distribution to every unknown parameter in the model. The sign indicators s_{ij} and order r_i are assumed uniform on the set $\{-1,1\}$ and $\{0,1,\ldots,R\}$, respectively, for the maximum order R. The interaction terms J_i and the components of interaction effects are also uniformly selected. For example, if two-way interaction $J_i=2$ is selected with predictors $\boldsymbol{X}_i = (X_{i1}, X_{i2}, X_{i3})$, then the interaction effect is equally likely to be one of $\{X_{i1}X_{i2}, X_{i1}X_{i3}, X_{i2}X_{i3}\}$. The prior for the number of knots and their locations are assumed to be uniform, respectively, on positive integers less than the number of training data points and on the location of training data points. The a_i use a Gaussian prior with mean zero and variance of 10^4. By using a class of reversible jump Metropolis–Hastings algorithms for Markov chain Monte Carlo (MCMC) [19], the BMARS collects samples of parameters from their joint posterior distribution. An advantage of BMARS is to identify significant main effects and interaction effects. In addition, the distribution of knots of each predictor reveals the complexity of the relationship between each predictor and the response.

4.2. Gaussian process regression

Gaussian process regression (GPR) generates a Gaussian distribution of functions that attempts to interpolate the output data. Specifically, the Gaussian process is a collection of random variables, where any finite subset of the random variables has a joint Gaussian distribution. The random variables for Gaussian process regression are the values of $f(\mathbf{X}_i)$ at the given points $\mathbf{X}_i$. Like a Gaussian distribution, a Gaussian process is entirely determined by its mean and covariance. The data are normally standardized to have a mean of zero, and the covariance is determined by a covariance function that is chosen to have certain properties. In our case we use the squared exponential covariance function which assures that the function $f(\mathbf{X}_i)$ is smooth. This covariance function has the form

$$k(\mathbf{X}_i, \mathbf{X}_j) = \sigma_f^2 \exp\left\{ -\sum_{k=1}^{m} \frac{(x_{ik} - x_{jk})^2}{2l_k^2} \right\}. \tag{4.4}$$

The covariance function has $m+1$ parameters: the maximum allowable covariance (σ_f^2), and the m length parameters l_k. These parameters are estimated using the empirical Bayes procedure [20,21]. Using the covariance function we construct a covariance matrix $\mathbf{K}$ of size $n \times n$ with elements

$$K_{ij} = k(\mathbf{X}_i, \mathbf{X}_j), \tag{4.5}$$

and then write the Gaussian process regression distribution for training data $\mathbf{X}$ as

$$\mathbf{f}(\mathbf{X}) \sim \mathcal{N}(0, \mathbf{K}). \tag{4.6}$$

Then to predict $\mathbf{f}(\mathbf{X}^*)$ at some number n^* of input points, we use the fact that $f(\mathbf{X})$ and $f(\mathbf{X}^*)$ are distributed by a joint Gaussian. Then the conditional expected value of $\mathbf{f}(\mathbf{X}^*)$ is

$$\mathrm{E}[\mathbf{f}(\mathbf{X}^*)|\mathbf{f}(\mathbf{X}), \mathbf{X}^*, \mathbf{X}] = \mathbf{K}(\mathbf{X}^*, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X})^{-1}\mathbf{f}(\mathbf{X}). \tag{4.7}$$

The covariance for $\mathbf{f}(\mathbf{X}^*)$ is given by

$$\mathrm{var}[\mathbf{f}(\mathbf{X}^*)|\mathbf{f}(\mathbf{X}), \mathbf{X}^*, \mathbf{X}] = \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) - \mathbf{K}(\mathbf{X}^*, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X})^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}^*). \tag{4.8}$$

Therefore, in Gaussian process regression we have a value for the mean and covariance at each point where we wish to evaluate the regression model.

4.3. Comparison of models

The two approaches, GPR and Bayesian MARS take two different approaches to the regression problem. GPR takes a holistic view of the data: it builds a regression model by evaluating a covariance function at every input point. This contrasts with the MARS approach that uses knot points to segregate the data into snapshots that can be described using different basis functions. As a result of these different perspectives, GPR can take a look at the overall effects of the data on the regression model, whereas MARS is formulated to look at both the main effects of the input data as well as their interactions. The approach of MARS is superior when the true function has different regimes. For example if it has a rapidly varying region and an asymptotic regime, as is the case with the function $f(x) = (\log x)^2/\sqrt{x}$, MARS can separate $f(x)$ into a rapidly varying piece near $x=0$ and a slowly varying piece as $x \to \infty$. Gaussian Process Regression cannot discriminate between these two different regimes of the function. We show results for this regression problem in Fig. 4. The results demonstrate the above point that GPR cannot capture the change in behavior of the underlying function. We do note, however, that modifications to the standard GPR such as the treed GPR [22] model can allow it to handle changes in function behavior. Also, an ANOVA decomposition can be used to determine interactions between inputs and other effects [23]. We plan on using these tools in future studies.

One drawback to the BMARS result that is obvious from Fig. 4 is that it still has some uncertainty at the training data. If the data have no uncertainty, as in the output from a computer model, the uncertainty that appears in the BMARS result is undesirable. Whereas the GPR results show where more training data is needed to eliminate uncertainty, the BMARS uncertainty away from training points is similar to that near training points. Another potential issue with BMARS is that the priors specified for the parameters of the fit may not be sufficiently broad to capture the correct effects. We attempt to deal with this by using very expansive priors, though we cannot be guaranteed that these are large enough for every application.

In terms of the number of parameters required to describe the regression model, Gaussian Process Regression requires only $m+1$ parameters. This contrasts with MARS, where there are parameters that describe the knot points, the interactions, the sign indicators, and basis function order. Of course, for the extra parameters one receives a more flexible regression model as discussed above. This is a two-edged sword, however, because the model can be over-parameterized by higher order interaction,

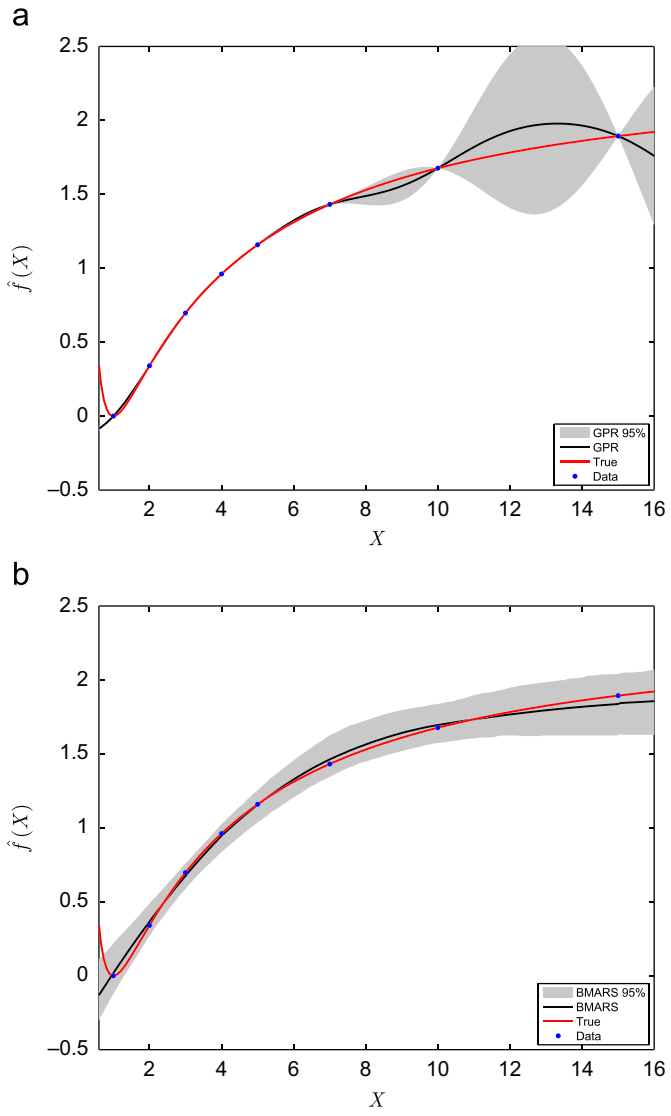

Fig. 4. Comparison of the two regression models under consideration, GPR and BMARS, for $f(x) = (\log x)^2/\sqrt{x}$: (a) GPR and (b) BMARS.

larger number of knots, or higher order basis functions. Therefore, after constructing a BMARS model one must look at the results to verify that these parameters have reasonable values.

5. Simulations

To map the relevant input space for our radiating shock simulation we have run 512 Hyades simulations. The input space we consider is 15 dimensional and we apply a uniform distribution for each dimension; see Table 2 for a list of the different input parameters and the range for each. Of these 15 different inputs seven describe the experimental configuration being simulated (the first seven entries in Table 2), and 8 are parameters that relate to numerical accuracy (e.g., number of zones in the Be disk) and model calibration (e.g., Be gamma in an ideal gas model for the equation of state). The 512 simulations used a Latin hypercube design to partition the 15 dimensional input space. See Fig. 5 for some scatter plots from this design.

The range of parameters related to the experimental configuration were determined from measured variances in the experiments combined with expert judgment. In all cases the range is

Table 2
Hyades simulation input parameters and ranges.

Parameter	Nominal	Range (%)	Min	Max
Beryllium thickness	0.020 mm	10	0.018	0.022
Laser energy	3.8 kJ	15	3.23	4.37
Pulse duration FWHM	1 ns	10	0.9	1.1
Xenon density	0.0065 g/cc	10	0.00585	0.00715
Tube length	5 mm	−20	4	5
Laser rise time	100 ps	50	50	150
Slope of laser pulse		30	0.85	1.15
Mesh resolution (number of zones in Be)	65 zones		20	200
Number of photon groups			20	100
Electron flux limiter	0.05		0.03	0.1
Time step control multiplier	1		0.25	1
Beryllium opacity multiplier	1		0.7	1.3
Beryllium gamma			1.4	1.667
Xenon gamma			1.2	1.4
Xenon opacity multiplier	1		0.7	1.3

selected to be larger than the true expectations of range. The Be thickness and tube length are based on the variance of fabrication measurements of the targets from the last several shots in the radiative shock campaign. The variances for the laser energy, pulse duration, rise time and pulse slope are determined from rough analysis of summary information after each experiment. Each of those measurements are known to a high degree of accuracy, but they vary from shot-to-shot by the amount indicated. The pressure of the Xe gas is recorded from a transducer readout right up to the shot and is also known very well. It can, however, vary from target-to-target due to varying leak rates and the time it takes from target delivery to shot at a given leak rate.

The code parameters were chosen after discussion of what constituted a reasonable and sensible range. In the case of the electron flux limiter, a parameter that attempts to correct for the fact that Hyades uses a diffusion model even though there are super-thermal electrons in the system for which this model is incorrect, the nominal value was chosen based on use of the code with similar experiments, and the range was set so that it encompassed the ranges used in other simulation studies [24–26]. The gamma and opacity multiplier (a parameter used to increase or decrease the opacity) for each material was chosen to represent what physical values were considered reasonable for the model. The amount the time step size is allowed to change, the so-called time step multiplier, was limited on the high end for code stability purposes and was limited on the low end mainly for practicality. Lastly, the mesh and photon group resolution nominal values were set based on use of the code with similar experiments. The range of both was set such that the top end would be well above the value necessary for convergence of the outputs and the low end would provide somewhat underresolved results in a detectable way.

In Fig. 6 the values for the shock location and density at the shock measured at 1.3 ns are shown as a function of the six input parameters that were shown to be most important by the sensitivity analysis below. In these scatter plots a discernible trend in the data as a function of an input parameter indicates that the output quantity of interest is strongly influenced by that parameter. Fig. 6 indicates that the shock location is dependent on the electron flux limiter and the laser energy: as each of these parameter values increases the shock location appears to increase. The density at the shock does not have such an obvious

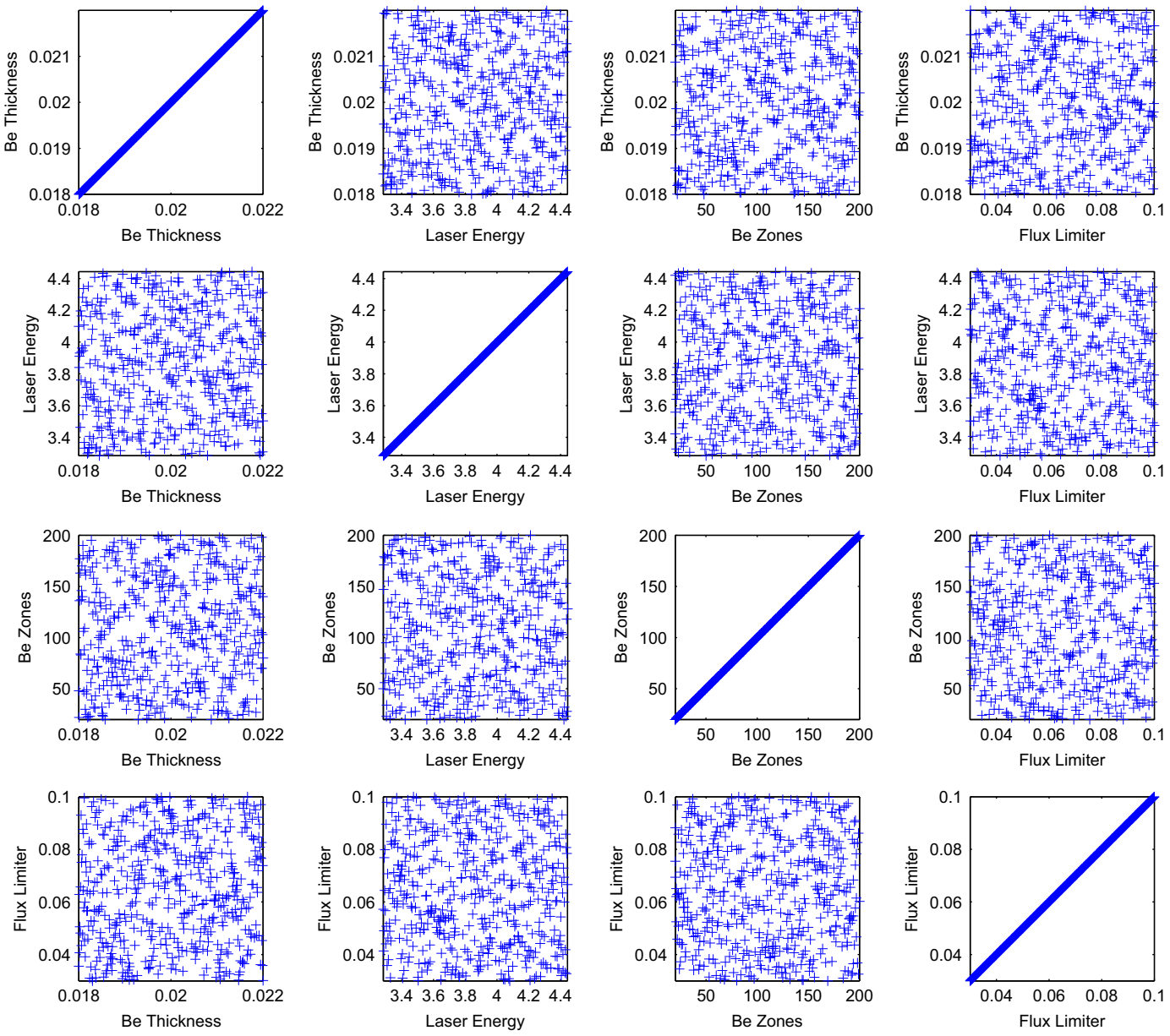

Fig. 5. Scatter plots for Latin hypercube design for 4 of the 15 input dimensions for building the Hyades emulator.

influence by any of these six parameters. Later, we will return to the question of which input parameters are most important by performing statistical analyses on the emulator results.

6. Hyades outputs emulation and discussion

The obvious litmus test for an emulator is how well it can predict the output of the code. To test the ability of GPR and BMARS to predict the output of Hyades we used 363 randomly selected Hyades runs from the set of 512 as test data. The results for shock position from BMARS and GPR emulators compared with the actual Hyades shock position on the test data are shown in Fig. 7. Shock position is one of the most important output parameters because it is experimentally measurable and the location of the shock in the initial conditions for CRASH should have a large effect on the CRASH output. In Fig. 7 perfect emulation would have the data fall on the red line given by $y=x$. A cursory glance at the figure shows that the mean values from BMARS did a better job of predicting the shock location than GPR, although it should be said that both regression methods did predict the shock position within to 3%. The emulator results for output parameters other than shock position demonstrated similar performance.

6.1. Analysis of GPR results

From the emulator we constructed using Gaussian Process Regression we can use the values of l_k found via the empirical Bayes method to get information on which input parameters affect the outputs the most. Specifically, the value of $1/l_k$, called the relative relevance, as computed by the empirical Bayes method provide some information on which input parameters affect the outputs the most [27]. Given that we standardized the inputs before building the GPR, the relative relevances are unitless. Relative relevance can under report the sensitivity due to

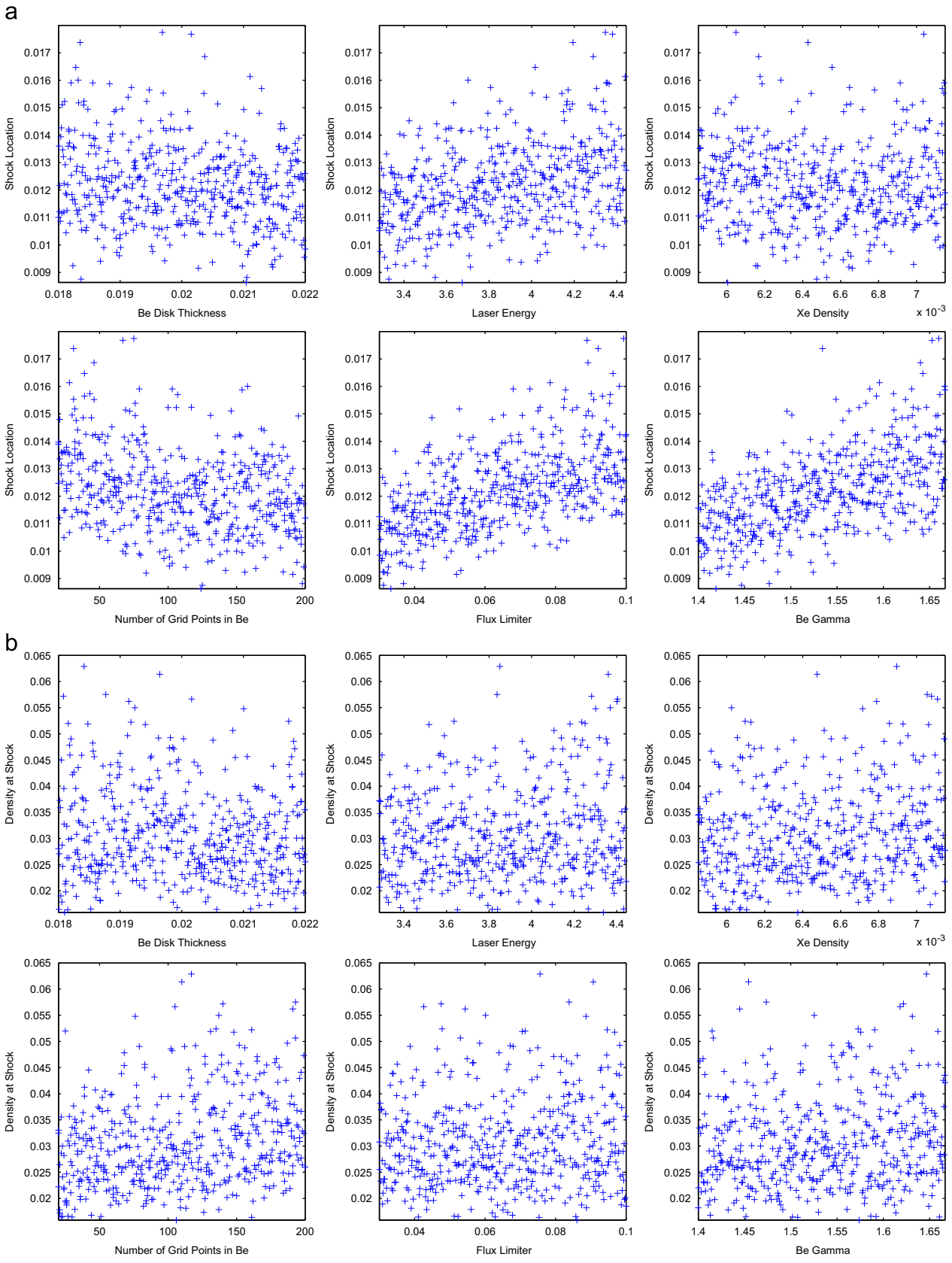

Fig. 6. Scatter plots for the shock location and density at the shock from the 512 Hyades simulations at 1.3 ns as a function of the six most important input parameters: (a) shock location and (b) density at shock.

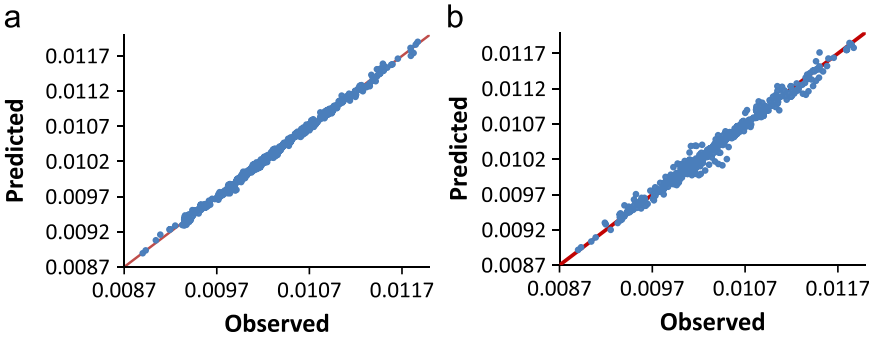

Fig. 7. Comparison of the shock position at 13 ns as predicted by the regression models with the observed Hyades value from the 363 test data results. For prediction we used the mean values from the models: (a) BMARS and (b) Gaussian process regression.

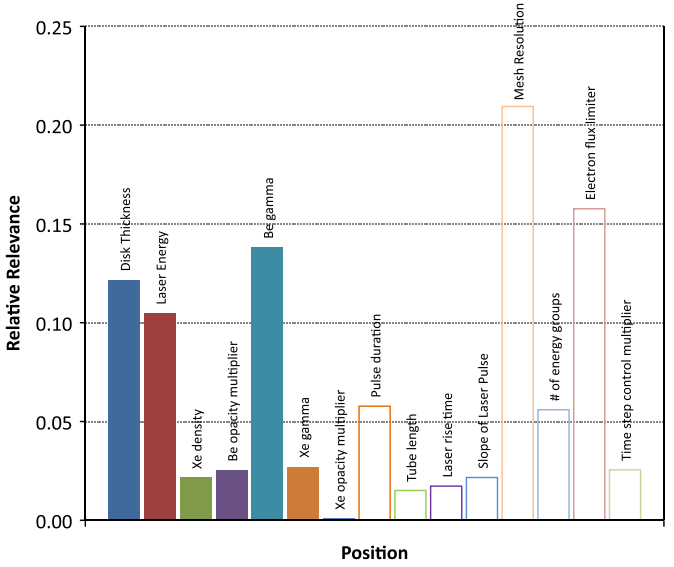

Fig. 8. Relative relevance ($1/l_k$) for each input parameter in the GPR emulator for the shock position at 13 ns.

strong linear interactions. To account for this we looked for strong linear effects in our outputs to assure that these effects are not missed. In the shock location output it appeared that there are linear effects in both the number of Be zones and the electron flux limiter (see Fig. 6). Nevertheless, as we will see below, the relative relevance for these terms is still significant.

In Fig. 8 we show the relative relevance for each input parameter, showing its influence on the shock position. From this figure we see that numerical and model calibration parameters, specifically the mesh resolution and the electron flux limiter, have the largest effect on the shock position. The fact that these parameters are important for the shock position is not surprising. Changing the number of mesh zones in the Be can change the shock position because the error in the numerical results is related to the mesh resolution as well as the fact that adding a Be zone might cause a discontinuous change in the shock position. In general, one would like to know the output when the number of zones is large to make the discretization error as small as possible. This consideration will be made when initializing CRASH.

Similarly, the electron flux limiter is a model parameter that attempts to account for the fact that a fluid model cannot properly capture nonlocal heat transport by the electrons. By changing this parameter the maximum rate at which heat is conducted by electrons to higher density from the laser-heated corona is changed. In turn changing how heat moves through the problem also changes how the shock moves. The large importance of the electron flux limiter has spurred us to further investigate the germane literature to properly constrain the range of this parameter.

Next in importance is the gamma of the Be material, a model calibration parameter that relates to the compressibility of the Be plasma. Experimental parameters, specifically the Be disk thickness and laser energy, are the fourth and fifth most important parameters. In turn adjusting the disk thickness and laser energy would have a larger effect on the shock position at 1.3 ns than, for instance, adjusting the laser rise time. The fact that parameters that describe the experiment are not the leading parameters in terms of relative relevance, indicates that the numerics and model calibration aspects of a Hyades simulation are the dominant mechanism for changing shock position.

The fact that the GPR results suggest that model calibration parameters are a critical part of controlling the shock position gets at the risk noted in our computational strategy. Given the fact that Hyades will have to be calibrated in these parameters, we will need to demonstrate that we can formulate a common setting or small range of settings for these parameters that will be adequate for predicting our year-five experiment. If we cannot demonstrate this we will be forced to (1) either account for this additional uncertainty in our predictions or (2) re-allocate resources to add a laser package to the CRASH code. In this way the GPR analysis is an example where statistical information will inform our the physics and modeling decisions.

6.2. Analysis of BMARS results

Using the BMARS results we have estimated the interactions between experimental parameters in the emulator. We do this by looking at the MCMC samples from the posterior distribution for the interaction parameters for the emulator and calculating the probability that a sample has that interaction term. In Fig. 9 we plot the probability that a particular interaction is in a sample of the emulator model for the shock position. The figure uses a 1 for laser energy, 2 for laser pulse duration, 3 for Xe density, and 4 for Be disk thickness; 0 denotes no interaction. For example, 100 denotes the effect of laser energy only, and 124 is the interaction between laser energy, laser pulse duration, and Be disk thickness. From the figure we can see that each of these four parameters is significant because each sample from the posterior distribution has these effects, i.e., we could not leave out these parameters and have an emulator of similar accuracy. Similarly, changing one of these parameters would change the shock position. Also, the two way interaction between laser energy and pulse duration and the three way interaction between laser energy, pulse duration, and Be disk thickness are important. These interactions are not completely unexpected. For instance, if the Be disk thickness is

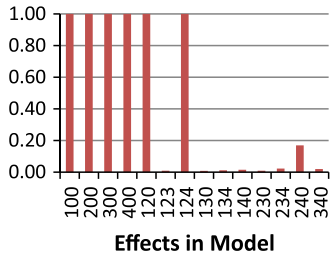

Fig. 9. Significance of effects of experimental parameters on the shock position. The effects are numbered 1=laser energy, 2=laser pulse duration, 3=Xe density, 4=Be disk thickness, and 0=no effect.

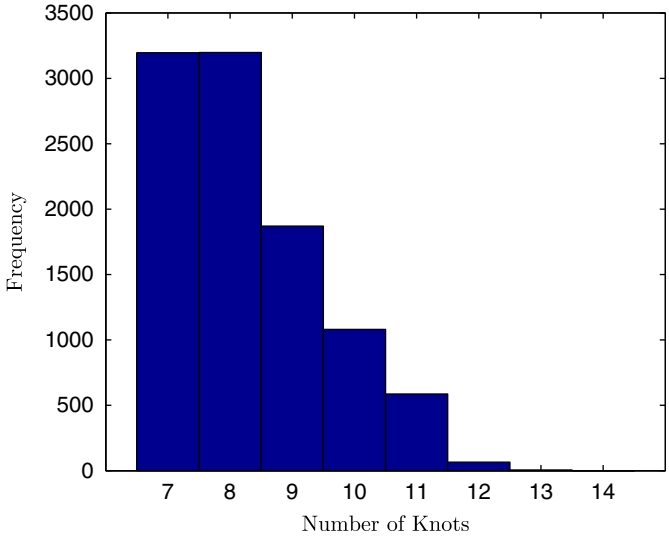

Fig. 10. Histogram for the number of knots in the BMARS emulator from samples of the posterior distribution.

increased, then the laser energy and pulse duration could be changed to compensate for the greater mass of Be to accelerate. This analysis, however, does not show what the relative strength of the interactions is, only what interactions are important.

From the BMARS emulator we can also estimate how many different "regimes" or snapshots of the input/output pairs are needed to predict the shock position. A histogram for the number of knots in the samples from the posterior distribution of the emulator for the shock position is shown in Fig. 10. From this figure we can see that the emulator needs at least 7 knots to describe the training data with 7 and 8 being the most common number of knots. Note that the training data allows up to 512 knot points.

7. Initialization of CRASH

The next step in generating CRASH simulations whose output could be used to assess its predictive capability was to use the emulator to initialize an uncertainty quantification run set of the CRASH code. The run set was based on a Latin hypercube design over 7 dimensions of the 15 dimensional input space and had 320 prescribed CRASH runs (see [8] in this issue for a detailed analysis of this study). To do this initialization we chose the GPR emulator because we were planning on constructing a GPR-based Kennedy-O'Hagan type model for the final simulation output relative to experiment [3].

To use the PIE to construct the initial state for the CRASH predictive model runs, we eliminate 8 of the Hyades inputs. For the number of energy groups and number of Be zones we evaluate the PIE at the 75th percentile of the input range because a study of the output showed the simulation to be well converged there. For laser pulse shape parameters (pulse duration, laser rise time, slope of the lase pulse) the nominal values are used because these parameters show only modest relative relevance to shock position and can be controlled more precisely than the input range to the study might suggest. Tube length is eliminated for a similar reason. The time step control multiplier also results in only a modest relative relevance and a swing in the range of the shock location of only 35 μm, compared to an average experimental error of 60 μm. The electron flux limiter value was also set to its nominal value. While this parameter shows significant effect on the shock location, we did not have a rational basis on which to calibrate it, and as shown in [8], the predictive model succeeds without it. We do note that picking the nominal values for some model calibration parameters would not be adequate to quantify the margin in a system's performance because the range over which those parameters vary introduce epistemic uncertainty into the CRASH results.

The results of generating the CRASH UQ run set using the GPR emulator for Hyades is shown in Fig. 11. Here we show the 40 PIE parameters as a function of input set number. Using the GPR emulator we have information about the distribution about the mean of the parameters. Later analysis of the CRASH outputs, perhaps by sampling from the distributions of the PIE parameters, will be able to discern if the uncertainty in the PIE parameters will impact the CRASH output.

8. Summary and conclusion

We have outlined a specific example of sensitivity analysis/uncertainty quantification in the case where functional output from one computer simulation is fed to another computer simulation. In our case both simulations are radiation-hydrodynamics calculations: the first simulation involves a calculation of the deposition of laser energy to a target, and the second models the long-time behavior of the radiating shock produced. To characterize the output of the first simulation, and thereby characterize the input to the second simulation, some sort of data reduction technique is necessary. Though purely statistical techniques exist to handle data reduction, we found they did not adequately reduce the degrees of freedom. We then resorted to physical insight about what features of the data were important.

Our efforts at physics-based data reduction were successful in regards to the fact that we were able to reduce the number of parameters required to describe the data from approximately thousands to 40. Using our reduced data we were then able to leverage existing nonparametric regression tools, namely Gaussian Process Regression and Bayesian MARS, to build an emulator for the simulation output and analyze the sensitivity in the mapping from input to output. This sensitivity analysis has lead us to further investigate how to reduce the uncertainty for several input parameters in order to reduce output sensitivity.

We believe that using experience and insight from a particular field goes hand in hand with the statistical techniques used in emulator construction and sensitivity analysis. Engineers and scientists should not abrogate their knowledge and experience when performing statistical analyses of simulation data. The statistical tools and statistics professionals can lead to enormous insight, but there are great synergies in combining statistical and engineering insight. Our results, we believe, demonstrate this. We encourage other researchers faced with data from numerical computations to use their knowledge of the science of the

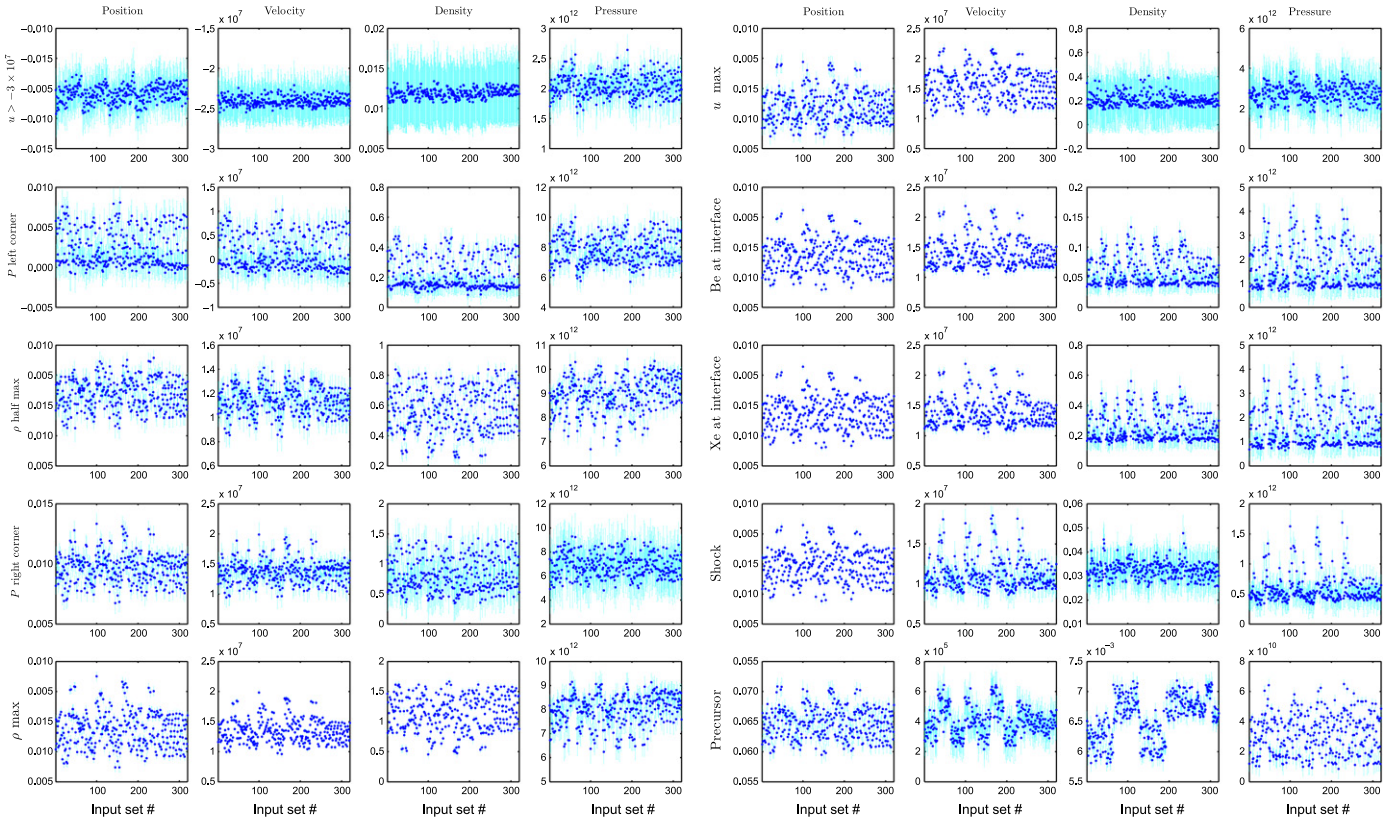

Fig. 11. Means and one standard deviation bounds of Hyades outputs to give 320 different initial conditions to CRASH.

underlying equations to aid them in reducing their data and in the larger mission of assessing predictive capability.

We believe that emulators are a useful tool in the uncertainty quantification piece of the QMU enterprise. In our study the emulator construction gave us insight into which inputs were the dominant mechanisms that influence a quantity of interest. An emulator can also be used in code calibration or to rapidly map the performance range of the system of interest, perhaps in creating an approximate cumulative distribution function for a quantity of interest. The emulator, however, should be used to inform where in input space to allocate computational resources for high fidelity, rather than make broad decrees about system performance.

We are presently working on a similar data reduction on two-dimensional functional data for two linked simulations. In this case we are trying to reduce tens or hundreds of thousands of parameters. For this case we believe it will be necessary to use both physics insight and statistical techniques to reduce the data to a manageable number of parameters. Overall, the lesson that physics insight is an essential part of the uncertainty quantification process has been instilled in our team.

Acknowledgments

This research was supported by the DOE NNSA/ASC under the Predictive Science Academic Alliance Program by Grant number DEFC52- 08NA28616. R.G. McClarren, D. Ryu, and B. Mallick's contributions were partially supported by Award no. KUS-C1-016-04, made by King Abdullah University of Science and Technology (KAUST).

References

[1] Committee on the Evaluation of Quantification of Margins and Uncertainties Methodology for Assessing and Certifying the Reliability of the Nuclear Stockpile. Evaluation of quantification of margins and uncertainties methodology for assessing and certifying the reliability of the nuclear stockpile. Washington, DC: National Academies Press; 2008.

[2] Higdon D, Gattiker J, Williams B, Rightley M. Computer model calibration using high-dimensional output. Journal of the American Statistical Association 2008;103(482):570–83.

[3] Kennedy M, O'Hagan A. Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2001;68:425–64. [with discussion].

[4] Higdon D, Kennedy M, Cavendish JC, Cafeo JA, Ryne RD. Combining field data and computer simulations for calibration and prediction. SIAM Journal on Scientific Computing 2004;26(2):448–66.

[5] Bayarri MJ, Berger JO, Cafeo J, Garcia-Donato G, Liu F, Palomo J, et al. Computer model validation with functional output. The Annals of Statistics 2007;35(5):1874–906.

[6] Powell KG, Roe PL, Linde TJ, Gombosi TI, De Zeeuv DL. A solution-adaptive upwind scheme for ideal magnetohydrodynamics. Journal of Computational Physics 1999;154:284–309.

[7] Larsen JT, Lane SM. HYADES: a plasma hydrodynamics code for dense plasma studies. Journal of Quantitative Spectroscopy and Radiative Transfer 1994;51:179.

[8] Holloway JP, Bingham D, Chou C-C, Doss F, Drake RP, Fryxell B, et al. Predictive modeling of a radiative shock system. Reliability Engineering and System Safety, this issue. doi:10.1016/j.ress.2010.08.011.

[9] Reighard AB, Drake RP, Dannenberg KK, Kremer DJ, Harding EC, Leibrandt DR, et al. Collapsing radiative shocks in xenon on the Omega laser. Physics of Plasmas 2006;13:082901.

[10] Michaut C, Falize E, Cavet C, Bouquet S, Koenig M, Vinci T, et al. Classification of and recent research involving radiative shocks. Astrophysics and Space Science 2009;322(1–4):77–84.

[11] Drake RP. Radiative shocks in astrophysics and the laboratory. Astrophysics and Space Science 2005;298:49–59.

[12] Bowers RL, Wilson JR. Numerical modeling in applied physics and astrophysics. Jones and Bartlett; 1991.

[13] Denison DGT, Holmes CC, Mallick BK, Smith AFM. Bayesian methods for nonlinear classification and regression. Wiley; 2002.

[14] Doss FW, Drake RP, Kuranz CC. Repeatability in radiative shock tube experiments. High Energy Density Physics 2010;6:157–61.

[15] Holmes CC, Denison DGT, Mallick BK. Accounting for model uncertainty in seemingly unrelated regressions. Journal of Computational and Graphical Statistics 2002;11(3):533–51.
[16] Friedman JH. Multivariate adaptive regression splines. The Annals of Statistics 1991;19(1):1–141. [with discussion].
[17] Craven P, Wahba G. Smoothing noisy data with spline functions. Numerische Mathematik 1979;31:377–403.
[18] Denison DGT, Mallick BK, Smith AFM. Bayesian MARS. Statistics and Computing 1998;8(4):337–46.
[19] Gilks W, Spiegelhalter D. Markov chain Monte Carlo in practice. Chapman & Hall/CRC; 1996.
[20] Carlin BP, Louis TA. Bayes and empirical Bayes methods for data analysis. Chapman & Hall; 2010.
[21] Neal RM. Regression and classification using Gaussian process priors. In: Bernardo JM, Berger JO, Dawid AP, Smith AFM, editors. Bayesian statistics, vol. 6. Oxford University Press; 1998. p. 475–501.
[22] Gramacy Robert B, Lee Herbert KH. Journal of the American Statistical Association 2008;103(483):1119–30. doi:10.1198/016214508000000689.
[23] Oakley J, O'Hagan A. Bayesian analysis of computer model outputs. In: Winkler J, Niranjan M, editors. Uncertainty in geometric computations, vol. 1. Springer; 2002. p. 119–130.
[24] Drake RP. High energy density physics. New York: Springer-Verlag; 2006.
[25] Sunahara A, Delettrez JA, Stoeckl C, Short RW, Skupsky S. Time-dependent electron thermal flux inhibition in direct-drive laser implosions. Physical Review Letters 2003;91:095003.
[26] Hu SX, Smalyuk VA, Goncharov VN, Skupsky S, Sangster TC, Meyerhofer DD, et al. Validation of thermal-transport modeling with direct-drive, planar-foil acceleration experiments on OMEGA. Physical Review Letters 2008;101:055002.
[27] Rasmussen CE, Williams CKI. Gaussian processes for machine learning. The MIT Press; 2006.