An initial exploration of Bayesian model calibration for estimating the composition of rocks and soils on Mars

Claire-Alice Hébert1 | Earl Lawrence2 | Kary Myers2 | James P. Colgan3 | Elizabeth J. Judge4

1Department of Applied Physics, Stanford University, Stanford, California, USA
2Statistical Sciences, Los Alamos National Laboratory, Los Alamos, New Mexico, USA
3Physics and Chemistry of Materials, Los Alamos National Laboratory, Los Alamos, New Mexico, USA
4Chemical Diagnostics and Engineering, Los Alamos National Laboratory, Los Alamos, New Mexico, USA

Abstract
The Mars Curiosity rover carries an instrument, ChemCam, designed to measure the composition of surface rocks and soil using laser-induced breakdown spectroscopy (LIBS). The measured spectra from this instrument must be analyzed to identify the component elements in the target sample, as well as their relative proportions. This process, which we call disaggregation, is complicated by so-called matrix effects, which describe nonlinear changes in the relative heights of emission lines as an unknown function of composition due to atomic interactions within the LIBS plasma. In this work, we explore the use of the plasma physics code ATOMIC, developed at Los Alamos National Laboratory, for the disaggregation task. ATOMIC has recently been used to model LIBS spectra and can robustly reproduce matrix effects from first principles. The ability of ATOMIC to predict LIBS spectra presents an exciting opportunity to perform disaggregation in a manner not yet tried in the LIBS community, namely via Bayesian model calibration. However, using it directly to solve our inverse problem is computationally intractable due to the large parameter space and the computation time required to produce a single output. Therefore, we also explore the use of emulators as a fast solution for this analysis. We discuss a proof of concept Gaussian process emulator for disaggregating two-element compounds of sodium and copper. The training and test datasets were simulated with ATOMIC using a Latin hypercube design. After testing the performance of the emulator, we successfully recover the composition of 25 test spectra with Bayesian model calibration.

KEYWORDS
ATOMIC, Bayesian calibration, disaggregation, emulation, Gaussian process, laser-induced breakdown spectroscopy, LIBS, modular calibration

1 | INTRODUCTION
One of the main scientific drivers of the Mars rover Curiosity is to determine whether Mars has ever been host to forms of life [16]. ChemCam, one of the instruments on board, developed by Los Alamos National Laboratory and L’Institut de Recherche en Astrophysique et Planétologie, is designed to record detailed data about the surface soil and rocks of the planet. The instrument uses laser-induced breakdown spectroscopy (LIBS) to measure the abundance
of all chemical elements by firing a laser onto a small patch of rock or soil surface, producing a plasma. As the high-temperature plasma cools, it emits light that ChemCam records via a spectrometer and CCD camera, producing a detailed spectrum over a range of wavelengths. Intensity peaks in such a spectrum can be used to identify the presence and relative abundance of chemical species in the sample of rock or soil. The absence or presence of certain elements could hold important evidence about the possibility of life on the rocky planet.

Curiosity can obtain hundreds of these spectra every day, but analyzing them remains a slow manual process. To successfully analyze, or disaggregate, each spectrum, we must answer two questions: what elements were present in the sample, and what are their relative proportions or abundances?

An expert can often answer the first question, identifying component elements by the presence of signature peaks in the spectrum at specific wavelengths. Estimating the relative abundance of these constituent species, however, can be quite difficult due to atomic interactions within the plasma. Due to these interactions, the relative heights of emissions lines of two elements can have a nonlinear dependence on the relative abundance of those species. These matrix effects—where matrix refers to the components of the target rather than to the mathematical concept—complicate our disaggregation problem: the spectrum of a multispecies target is not simply the linear combination of the spectra for each individual element. These nonlinear effects pose significant challenges for disaggregation of LIBS data, particularly because we do not have a closed form expression of the effects.

Clegg et al. [4] introduced the use of multivariate approaches, such as partial least squares regression, to improve the ability to determine the elements present in a sample and estimate their relative proportions in the presence of matrix effects. More recently, scientists at Los Alamos National Laboratory adapted a first-principles plasma physics code called ATOMIC [15] to provide a forward model for the emission from LIBS plasmas, including their matrix effects [6, 7, 11]. The new existence of this forward model presents an exciting opportunity to explore the use of Bayesian model calibration [12] to compute estimates, with associated uncertainties, of the components of a target and their relative abundances. That is, we want to solve the inverse problem to determine the input parameters for ATOMIC, which include the elements present and their proportions that produce the simulated spectrum that is most like an unknown measured spectrum.

Bayesian model calibration in the context of high-dimensional outputs [10], of which LIBS spectra are an example, has a successful history in a variety of scientific applications. For example, in materials science, these methods have been applied to estimate strength parameters of aluminum alloys in hydrodynamic shock experiments [19]. Closer to our application, statistical and machine learning methods have also been explored as a way to solve the inverse problem in spectroscopy; recent examples are the use of generalized linear models and neural networks to diagnose magnetic fields in plasmas [3] and the application of approximate Bayesian computation to radioisotope identification in gamma-ray spectroscopy [2].

We draw inspiration from Judge et al [11], who demonstrated the impact of matrix effects on sodium peak heights in a simple two-element mix of sodium and copper measured with LIBS. The authors also demonstrated the ability of ATOMIC to replicate those effects from first principles. In particular, their experimental observations, supported by ATOMIC’s theoretical calculations and modeling, showed that the sodium lines increased significantly in emission intensity as more copper was added to the target. This effect is explained by an increase in electron density, due to the copper, leading to increased recombination within the plasma.

In this paper, we will also use simplified two-element targets of sodium and copper to begin to explore our ability to use Bayesian model calibration with ATOMIC to perform disaggregation. As we look forward to more complex targets drawn from the large parameter space of all possible elements, and particularly in the context of the high data-collection rate of ChemCam on Curiosity, we recognize that using ATOMIC directly in this framework will be computationally intractable. We, therefore, also present our results building and evaluating emulators to provide fast approximations to the computationally expensive ATOMIC runs.

Emulators are well-established tools in the context of slow computer models and their use for modeling spectra has been demonstrated in the field of cosmology. Large N-body simulations of the universe are prohibitively expensive, and emulating the matter power spectrum of the universe on cosmological scales was shown to be an effective solution [13]. Although these spectra have very different physical origins from those of LIBS, these prior results provide some context for our efforts presented here.

Our team’s ongoing and preliminary work [1] includes success at estimating the plasma temperature and density...
of simple, fixed compounds, taking into account a structured discrepancy between ATOMIC simulations and measured LIBS data. While Judge et al. [11] showed that ATOMIC can replicate the matrix effects in experimental data, the discrepancy study in Bhat et al. [1] and our own explorations showed some as yet unresolved challenges when comparing ATOMIC and measured LIBS spectra in the quantitative way required to support Bayesian model calibration. Therefore, we focus here on the methodology of performing disaggregation in the presence of matrix effects in a simplified scenario, using a test set of simulated ATOMIC spectra rather than of measured spectra as would be our ultimate intent. This allows us to demonstrate the feasibility of using modular Bayesian model calibration to perform disaggregation with LIBS data.

We present an overview of the ATOMIC simulations, inputs, and outputs in the following section. In Section 3 we outline the statistical framework of emulation and Bayesian calibration used in this work, and we discuss their application to simulation data in Section 4. We conclude with further discussion of the relevance and context of our results and future directions.

2 | PLASMA SIMULATIONS

Computer model calibration relies on a small number of runs of a high-fidelity simulation, tiled across parameter space. This set of simulations can then be used to build emulators, which can quickly approximate the computer model at new parameter settings. In the context of the analysis of LIBS data, the model of interest is the ATOMIC forward model, a general purpose plasma modeling and kinetics code. It was developed to simulate the emission spectra of chemical compounds using first principles theoretical atomic physics [15], in particular, emission or absorption spectra from plasmas either in local-thermodynamic equilibrium (LTE) or in non-LTE.

ATOMIC simulations require a few primary inputs: the temperature and density of the plasma, and a model describing the atomic structure and scattering data of the material(s) constituting the plasma. These last, which include quantities such as energy levels and transition probabilities, are generated from the Los Alamos suite of atomic physics codes [9]. The simulations discussed here [5] were generated from the CATS code [8] with modifications made for plasmas generated from LIBS [7]. For a given temperature and density, ATOMIC then models the emissivity of the plasma by computing its average ionization.

As motivated above, the simulations discussed in this paper are of sodium copper (NaCu) plasmas with varying ratios of sodium (Na) to copper (Cu), inspired by Judge et al. [11]. In addition, we vary the input plasma temperature $T$ and the mass density $\rho$ of the plasma. In particular, we run simulations over the following space of these three parameters:

1. plasma temperature $T$, in units electron volts (eV) and range [0.5, 1.5]
2. mass density $\rho$, in units of g/cm$^3$ and range [$-7$, $-4$] on the log$_{10}$ scale
3. composition of the plasma, defined as the proportion of one of the two elements in our compound. We have arbitrarily chosen to use the proportion of sodium in the plasma, $\%$Na, in the range [0, 1]. $\%$Cu can be retrieved by using $1-%Na$.

A few comments are in order about the third input parameter, $\%$Na. It is by solving the inverse problem for this parameter that we are doing disaggregation—that is identifying the elements present and estimating their proportions. The validity of this particular parameterization, where $\%Cu = 1-%Na$, holds only because our simulations are run in an artificially simplified scenario without atmosphere, components of which, such as carbon and oxygen, would usually be present in the plasma in unknown quantities. Thus, we know a priori that the only elements that could be present are sodium and copper. This enormously simplifies the first part of the disaggregation problem: identifying which elements are present. Our thinking is that success in this very constrained regime will establish a foundation for addressing the challenges faced by ChemCam on Mars, such as the presence of atmosphere and much larger sets of candidate elements.

The output of the ATOMIC simulation is a spectrum with intensity as a function of wavelength for a particular set of the above inputs, that is, a spectrum for a particular compound at some plasma density and temperature. The simulation provides intensity as power per volume per photon energy per unit solid angle. ATOMIC predictions do not account for any effects arising from the spectrometer, but do include matrix effects. The computation time for a single ATOMIC run depends on the chemical complexity of the compound of interest, and typically varies from minutes to hours on a high-performance computing system.

The ATOMIC simulations used in this analysis were selected using two Latin hypercube designs over the three input parameters: a training set of 500 simulations, and an independent test design of 25 points to which we added noise before analysis, as described in Section 4.2. The training set parameter design, as well as a few of the resulting spectra, are shown in Figure 1. We have labeled the peak locations for two sodium peaks and one copper peak along the top wavelength axes, and will show these markers.
FIGURE 1 Three examples of ATOMIC simulations used for training along with the design for the training set. Left panel: three example training spectra, with intensity in units of power per volume per photon energy per unit solid angle. Middle: The same three examples plotted as log intensity, thereby reducing the orders of magnitude variation across the data. Right panel: simulation input design for the 500 training spectra, scaled to the [0,1] range for all parameters. The parameter settings for the three examples shown in the previous panels are highlighted in color.

throughout this work when relevant. These are not the complete set of peaks for sodium or copper and are simply meant to provide some reference to the eye.

We note, in the left panel of Figure 1, the orders of magnitude over which the simulation output varies. This motivates a rescaling to the log scale, as shown in the middle panel, before statistical analysis, due to anticipated difficulty capturing small, yet potentially important, details in the presence of these large variations in amplitude. While this log transform is not standard practice in the spectroscopy community, exploratory analysis confirmed that emulators built for the spectra on their original scale performed significantly worse than the results presented here. Additional motivations for modeling on the log scale are discussed in Bhat et al. [1].

3 | MODULAR BAYESIAN CALIBRATION

The aim of this work is to demonstrate the application of Bayesian calibration methods to the problem of disaggregating LIBS spectra—identifying the elements present and estimating their proportions. We start with a brief overview of computer model calibration from Kennedy and O’Hagan [12], as it applies to our simplified scenario of simulated test data. Computer model calibration entails estimating the input parameters of a computer model (here, ATOMIC) that most likely generated some given observed data (here, these data are simulated observations).

Let $y$ be an observed spectrum. A key assumption is that $y$ is a noisy version of the simulator output at some unknown parameter setting $\theta$, with no discrepancy:

$$y = \eta(\theta) + \epsilon$$  \hspace{1cm} (1)

We will assume that the data $y$ have been centered and scaled according to the mean vector $\mu$ and scalar standard deviation $\sigma$ of the training data.

Following the well-established literature on calibration, we denote the ATOMIC computer model as $\eta(t)$, which takes a generic $p$-dimensional parameter vector $t$ as input to produce a LIBS spectrum. Here, $p = 3$ for the three input parameters described in Section 2. The vector $\theta$ represents the specific values of $t$ that yield the model output $\eta$ that most closely resembles an observed spectrum, as in Equation (1). Note that we are working here with $y$ and $\eta(t)$ as log scaled versions of the measured and modeled spectra.

We estimate $\theta$ for a given observation $y$ by exploring the posterior $p(\theta|y)$ with Markov chain Monte Carlo (MCMC). This posterior is calculated, via Bayes rule, as proportional to the product of the data likelihood $p(y|\theta)$ and the parameter prior $p(\theta)$. We will first define the data likelihood, and the choice of prior is discussed below. Equation (1) describes how the data are generated given the parameters $\theta$. The noise $\epsilon$ determines the sampling function for the data. Here we take $\epsilon$ to be normally distributed with mean 0 and variance $\Sigma_y$, which gives the following likelihood:

$$y \mid \theta \sim \mathcal{N}(\eta(\theta), \Sigma_y)$$  \hspace{1cm} (2)

This is in principle all we need (along with a prior) to perform MCMC. There is a significant computational challenge, though, as the ATOMIC model $\eta$ is slow to compute: given some vector $t$, evaluating $\eta(t)$ takes order...
of minutes or hours, rendering exploration via MCMC extremely slow. We will follow the standard approach to overcome this through use of an emulator: a statistical model that provides a fast approximation of the simulator output.

In the following subsections we discuss first the approach of emulating the ATOMIC outputs using Gaussian processes, then details of Bayesian model calibration.

### 3.1 Gaussian process emulation

As mentioned above, we define a statistical model to provide fast approximations of the slow ATOMIC outputs. This emulator will be some unknown function conditioned on a training set of \( m \) simulator runs \( \{\eta(t_1), \ldots, \eta(t_m)\} \) at fixed inputs \( t_1, \ldots, t_m \). For simplicity, we scale these inputs such that \( t \in [0, 1]^p \). Before we describe the details of the emulator, we address a second computational bottleneck due to the high-dimensionality of the data. Each simulator output \( \eta \) has \( n_\eta = 32,000 \) wavelength bins. Naive MCMC implementations might require the inversion of a \( 32,000 \times 32,000 \) matrix or \( 32,000 \times m \) matrices at each step to calculate the posterior. The solution to this problem, as developed in [8], relies on using dimensionality reduction to find a reduced set of basis vectors. This set of \( n_\eta \)-dimensional vectors \( \{k_i, i = 1, \ldots, q\} \) describes the model for any input \( t \) in the following way:

\[
\eta(t) = \sum_{i=1}^{q} k_i w_i(t) \quad (3)
\]

where the weights \( w_i(t) \) hold the dependence on the input \( t \). We denote the number of components included in the reconstruction by \( q \), with a maximum value of \( m \), the size of the training set. For this application, we found that \( q \sim 10 \) sufficed for good emulator performance. This dimensionality reduction formalism reduces the computational complexity of the problem enormously: we can now emulate and sample just \( q \sim 10 \) weights instead of \( n_\eta \sim 10^5 \) wavelength bins.

We find this new basis via a singular value decomposition (SVD) of the training simulation matrix \( X \). Each of the \( m \) columns of \( X \) holds a simulation of length 32,000. The SVD factorization of \( X \), in terms of orthogonal matrices \( U, V \) and diagonal matrix \( S \) of singular values of \( X \), can be written:

\[
X = USV^T = KW \quad (4)
\]

The second equality relates the SVD to Equation (3) in matrix form, with \( K \) a column matrix of all the \( k_i \)s and the weights \( w \) in row vector \( W \). We have defined \( K = US/\sqrt{m} \) and \( W = V^T/\sqrt{m} \).

In terms of the data described in Section 2, this decomposition results in a set of principal components \( K \), each of which has length \( n_\eta = 32,000 \) that are common between all the spectra, and a set of weights, which now hold the dependence on the input parameters, and which vary between each spectrum. The first four of these principal components (PCs) are shown in Figure 2 along with the associated weights for the 500 training and 25 test simulations.
Our ATOMIC emulator will take the form of Equation (3), with $K$ calculated from the set of training simulations. The weights $w_i$ define a surface in parameter space for each component $i$ (see Figure 2): as is common in the literature, we will represent each of these by a Gaussian process (GP):

$$w_i(t) \sim \mathcal{N}(0, \sigma_{wi}^2 R_i(t))$$

(5)

where $\sigma_{wi}^2$ is the marginal variance for weight $i$ and $R_i(t)$ is a correlation matrix with each entry given by the correlation function:

$$\text{corr}(t, t'; l_j) = \prod_{j=1}^{3} \exp \left( -\frac{|t_j - t'_j|^2}{2l_j^2} \right)$$

(6)

where $l_{ij}$ is the length scale hyperparameter for weight $i$ and parameter $j$.

Whereas it is common to perform the estimation of the GP hyperparameters concurrently with the calibration, we treat these in a modular way as in [14], fixing the hyperparameters by maximum likelihood estimation and keeping them fixed during the Bayesian calibration. This approach greatly speeds up the estimation process because we do not need to rebuild and reinvert the covariance matrices at each step of the MCMC. During calibration, we use the GPs to predict the value of each weight for parameter settings not present in the training simulations. The prediction of weight $i$ for some parameters $\theta$ can be found using properties of conditional normal distributions:

$$\hat{w}_i | \theta \sim \mathcal{N}(r_i(\theta)R^{-1}_i w_i, \sigma_{wi}[1 - r_i(\theta)^T R^{-1}_i r_i(\theta)])$$

(7)

Here $r_i(\theta)$ is the $m \times 1$ vector found by applying the GP covariance function in Equation (6) (with hyperparameters for weight $i$) to $\theta$ and the set of training simulation parameters $\{t_1, \ldots, t_m\}$. The $m \times m$ matrix $R_i$ is the correlation matrix of the training parameters, and $w_i$ is the $m \times 1$ vector of training weights for principal component $i$.

3.2 Model calibration

We now describe the process of estimating the input parameters of the simulations via Bayesian model calibration. As seen in Equation (2), we assume our data $y$ is given by a simulator run $\eta(\theta)$, with covariance $\Sigma_y$. Based on the Poisson nature of the spectra we expect that errors are independent across wavelengths bins, implying a diagonal $\Sigma_y$. We assume a fixed diagonal form, which can be parameterized by a precision $\lambda_y$: $\Sigma_y = \lambda_y^{-1} I$.

$$y | \eta(\theta) \sim \mathcal{N}(\eta(\theta), \lambda_y^{-1} I)$$

(8)

We have seen in the previous section how to express our data in a new basis of PCs $K$ and parameter-dependent weights $W$. The emulator output $\eta(\theta)$, given parameters $\theta$, is expressed in the new basis using the weights predicted by the GPs (see Equation (7)):

$$\eta(\theta) = \tilde{K} \hat{w}(\theta)$$

(9)

To cast a new observed spectrum $y$ into this basis, we use:

$$w_{\text{obs}} = \tilde{K} y$$

(10)

where $\tilde{K} = (K^T K)^{-1} K^T$, and we define $w_{\text{obs}}$ as the vector of $q$ weights corresponding to the observation $y$.

We use Equations (8)–(10) and properties of Gaussian distributions to find the sampling of the observed weights given $\hat{w}(\theta)$, the GP predictions:

$$w_{\text{obs}} \mid \hat{w}(\theta) \sim \mathcal{N}(\tilde{K} \hat{w}(\theta), \tilde{K} \lambda_y^{-1} I \tilde{K}^T)$$

(11)

$$= \mathcal{N}(\hat{w}(\theta), (\lambda_y K^T K)^{-1})$$

(12)

Finally, since each predicted value $\hat{w}_i$ is drawn from a GP, these weights are normally distributed given parameters $\theta$, as in Equation (7). This means that the observed weights $w_{\text{obs}}$, given some parameter vector $\theta$, are given by:

$$w_{\text{obs}} \mid \theta \sim \mathcal{N}(\mu_w, (\lambda_y K^T K)^{-1} + \Sigma_w)$$

(13)

where $\mu_w$ is a vector with entry $i$ given by $r_i(\theta) R^{-1}_i w_i$, $i = 1, \ldots, q$ and $\Sigma_w$ is a $q \times q$ diagonal matrix with element $ii$ given by $\sigma_{wi}[1 - r_i(\theta)^T R^{-1}_i r_i(\theta)]$, where the $i$ denotes that these are calculated from the covariance matrix with hyperparameters for weight $i$. Equation (13) is the likelihood we will use for MCMC exploration.

4 RESULTS AND DISCUSSION

We applied the methods above using the Python Scikit-learn implementation of GPs [17] and the Monte Carlo sampling in PyMC3 [18]. We chose a value of $q = 15$ PCs in the emulator used to generate all the results discussed in this section. The SVD reconstruction with these 15 components explains $>99.995\%$ of the variance of the training set. In addition, the emulator and calibration errors did not significantly improve with added components beyond this point. Indeed, when a larger value is chosen for $q$, the additional weights added are less and less constraining for calibration since their associated sampling variances (see the likelihood in Equation (13)) increase with weight index. Adding more weights, then, is not expected to result in more accurate calibration results.
Both emulation performance and calibration results are calculated based on a test set of 25 ATOMIC simulations run on a Latin hypercube design independent from the training set, to which we added noise.

### 4.1 Emulator performance

Figure 3 summarizes the emulator performance for the settings listed at the start of this section. The performance is calculated on 25 test simulations.

We quantify emulator performance by two metrics: $R^2$ and percent error, both calculated point-wise for each wavelength modeled. We calculate these by:

- $R^2 = 1 - \frac{\sigma^2_{\text{res}}}{\sigma^2_{\text{raw}}}$, where $\sigma^2_{\text{raw}}$ is the variance of the test simulations around their mean, and $\sigma^2_{\text{res}}$ is the variance of the residuals (emulator output - truth).

- $\% \text{error} = 100 \times \frac{\sigma(\eta(\theta) - y)}{\sigma y + \mu}$, calculated for each test example and where $\mu, \sigma$ are the mean vector and scalar standard deviation of the training set.

Overall, these errors lie between ±1%, with the exception of the test run plotted in yellow, which we discuss below. We conclude that the emulator is performing well.

We expect that the information most important for our eventual goal of disaggregation should lie near the peaks associated with constituent elements. For example, the locations of some important sodium and copper peaks are indicated in the left panel of Figure 3 by tick marks and vertical dashed lines. We see that the percentage error is greater at peak locations than at most other points along the wavelength axis. The dataset contains a wide range of sodium concentrations, implying high data variance at these peak locations. In absolute terms, the percent error...
shows that this variation is harder for the emulator to capture fully. However, the spikes in $R^2$ value at those locations are not significantly worse than at many other wavelengths, indicating that the emulator is performing well relative to the variance of the data. In summary, the emulator is able to capture important variations in the data, which we expect to be indicative of sodium concentration.

The right panel in Figure 3 shows the test design, each point colored according to the median percentage absolute error of the emulator predictions. The test run in yellow stands out as having the highest median error by far, as it dominates the color scale. Given its location on the very edge of parameter space for all three inputs, it is not surprising that the emulator has trouble with this particular test example. The percentage error of this test run is highlighted in yellow in the left hand panel, and we will discuss it further below. The median absolute emulator errors for the remaining 24 test runs are all at the low end, however, clustered in the $0.02 - 0.07$ percent range.

In summary, both the percentage errors and the $R^2$ values of the emulator predictions are indicative of strong performance, so we proceed with calibration.

### 4.2 Calibration results

The Bayesian calibration was performed for noise-added versions of the 25 test spectra, following Equation (8) with precision $\lambda_y = 4$. In addition to the 25 two-element NaCu compounds with varying proportions of Na, we also ran the calibration for 50 spectra of single-element simulations, with the composition for 25 fixed at 100% and 25 fixed at 0% Na. These single-element simulations were run with the same input design for plasma temperature and density as the 25 two-element test simulations. During analysis these single-element spectra were treated identically to the two-element test examples.

The summary of these calibration results is shown in Figure 4. The left three panels of this figure show very encouraging calibration results for the two-target simulations, with the $1\sigma$ uncertainties (shown by vertical bars) containing the truth in almost all cases. The leftmost panel shows successful disaggregation for all 25 test spectra. The points are colored according to the emulator errors as in the right panel of Figure 3; of particular interest is the test run with highest emulator error, discussed above, indicated in yellow. Recall that its true parameters are the highest %Na in the design, relatively high temperature $T$, and very low density $\rho$. We find that the disaggregation results for this challenging test run are on par with the rest of the test set.

Our single-element simulation results are summarized by the histograms in the last panel of 4. Our success in recognizing pure Na and pure Cu demonstrates the ability of the emulator trained on the two-element NaCu simulations to correctly identify examples of single elements. We had had some concern that the difference between a target
with a vanishingly small amount of one element and one with none at all would look like a discontinuity of some sort that would be difficult to capture with our emulators, and these results suggest that our emulators are robust to that difference.

Figure 5 shows bivariate marginal distributions of the MCMC samples for two-element test examples. The first was chosen to be a “typical” example from the test set, that is, low emulator error, and the second example shown is the high emulator error run, discussed previously. The top panel shows the two log scaled spectra, colored according to median absolute percent error, as discussed in Section 4.1. The middle and bottom panels of Figure 5 show the MCMC samples for the typical and high error examples, respectively. The subplots along each diagonal show the marginalized posteriors and true parameter values; calibration succeeds well on typical examples in the test set, and even when the emulator does relatively poorly, the calibration produces acceptable posteriors. All parameter combinations show some evidence of correlation between the samples, most notably between temperature and density in the bottom panel.

5 CONCLUSIONS

In this work, we show promising first results toward the goal of disaggregating LIBS data from ChemCam with Bayesian model calibration. The emulation and Bayesian calibration methods discussed here successfully perform disaggregation in the presence of matrix effects in noise-added simulations of two-element compounds.

An important next step is to test this approach on measured LIBS data. There are a number of challenges to overcome before this can succeed. While ATOMIC simulations compare very well to experimental data on many metrics and for other applications [6, 7, 11], the wavelength-by-wavelength matching that drives the emulation in this work is not one of them. For example, we have found that the peak shapes, widths, and to some extent even locations, differ between ATOMIC outputs and measured LIBS spectra. We are working with plasma physicists to better understand the discrepancy, as these small effects have not been straightforward to accommodate with the methods presented here.

In addition, LIBS spectra measured “in the wild” on Mars or even in a laboratory setting typically contain extraneous peaks that do not provide information about the target of interest. For instance, when measured in air, the plasmas include contributions from the atmosphere in unknown proportions that appear as peaks in the spectra. And laboratory targets of even simple two-element compounds like NaCu typically are formed using binders such as stearic acid that introduce still more peaks when measured via LIBS. These extraneous components of the plasmas exacerbate matrix effects and complicate our analysis. A possible path forward that we are exploring is to limit the set of wavelengths in our analyses to those around expert-identified peaks of interest, rather than considering the entire spectrum shown in the work here.

In addition, instrument response and other effects should be accounted for when comparing simulations to measurements. Standard practice for ChemCam data is to correct for some of these effects, and we have begun exploring how these corrections impact our ability to perform disaggregation.

While we continue those explorations toward comparing simulations with measurements as a separate line of research, we will build on our successes with simulated data that we presented here in order to provide a proof of principle for disaggregating increasingly complex multi-element targets.

ACKNOWLEDGMENTS

Research presented in this paper was supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under project number 20180097ER. CAH was supported by the Department of Energy Computational Science Graduate Fellowship (DE-FG02-97ER25308).

CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in ATOMIC Simulations and Experimental Data for Sodium and Copper Mixtures at https://doi.org/10.25583/1616321, reference number 5.

ORCID
Claire-Alice Hébert https://orcid.org/0000-0002-7397-2690

REFERENCES
1. K. S. Bhat, K. Myers, E. Lawrence, J. Colgan, and E. Judge, Structured discrepancy in Bayesian model calibration for ChemCam on the Mars Curiosity rover, Ann. Appl. Stat. 14(4) (2020), 2020–2036. http://dx.doi.org/10.1214/20-aoas1373.
2. T. Burr, A. Favalli, M. Lombardi, and J. Stinnett, Application of the approximate Bayesian computation algorithm to gamma-ray spectroscopy, Algorithms. 13(10) (2020), 265. https://doi.org/10.3390/a13100265, https://www.mdpi.com/1999-4893/13/10/265.
3. M. Cianciosa, K. J. Law, E. H. Martin, and D. L. Green, *Machine learning for analysis of atomic spectral data*, J. Quant. Spectrosc. Radiat. Transf. 240 (2020), 106671.

4. S. M. Clegg, E. Sklute, M. D. Dyar, J. E. Barefield, and R. C. Wiens, *Multivariate analysis of remote laserinduced breakdown spectroscopy spectra using partial least squares, principal component analysis, and related techniques*, Spectrochim. Acta B: At. Spectrosc. 64(1) (2009), 79–88.

5. J. Colgan, et al. *ATOMIC simulations and experimental data for sodium and copper mixtures*, 2020. https://doi.org/10.25583/1616321.

6. J. Colgan, E. Judge, H. Johns, D. Kilcrease, J. Barefield II, R. McInroy, P. Hakel, R. Wiens, and S. Clegg, *Theoretical modeling and analysis of the emission spectra of a ChemCam standard: Basalt BIR-1A*, Spectrochim. Acta B: At. Spectrosc. 110 (2015), 20–30.

7. J. Colgan, E. Judge, D. Kilcrease, and J. Barefield II, *Ab-initio modeling of an iron laser-induced plasma: Comparison between theoretical and experimental atomic emission spectra*, Spectrochim. Acta B: At. Spectrosc. 97 (2014), 65–73.

8. R. D. Cowan, *The theory of atomic structure and spectra*, Univ of California Press, Berkeley, 1981.

9. C. Fontes, H. Zhang, J. Abdallah Jr., R. Clark, D. Kilcrease, J. Colgan, R. Cunningham, P. Hakel, N. Magee, and M. Sherrill, *The Los Alamos suite of relativistic atomic physics codes*, J. Phys. B: At. Mol. Opt. Phys. 48(14) (2015), 144014.

10. D. Higdon, J. Gattiker, B. Williams, and M. Rightley, *Computer model calibration using high-dimensional output*, J. Am. Stat. Assoc. 103(482) (2008), 570–583. https://doi.org/10.1198/016214507000000888.

11. E. J. Judge, J. Colgan, K. Campbell, J. E. Barefield, H. M. Johns, D. P. Kilcrease, and S. Clegg, *Theoretical and experimental investigation of matrix effects observed in emission spectra of binary mixtures of sodium and copper and magnesium and copper pressed powders*, Spectrochim. Acta B: At. Spectrosc. 122 (2016), 142–148. https://doi.org/10.1016/j.sab.2016.06.004.

12. M. Kennedy and A. O’Hagan, *Bayesian calibration of computer models*, J. R. Stat. Soc. Ser. B (Stat. Methodol.) 63(3) (2001), 425–464.

13. E. Lawrence, K. Heitmann, J. Kwan, A. Upadhye, D. Bingham, S. Habib, D. Higdon, A. Pope, H. Finkel, and N. Frontiere, *The mira-titan universe. II. Matter power spectrum emulation*, Astrophys. J. 847(1) (2017), 50. https://doi.org/10.3847/1538-4357/aa86a9.

14. F. Liu, M. Bayarri, J. Berger, et al., *Modularization in Bayesian analysis, with emphasis on analysis of computer models*, Bayesian Anal. 4(1) (2009), 119–150.

15. N. H. Magee, J. Abdallah, J. Colgan, P. Hakel, D. Kilcrease, S. Mazevet, M. Sherrill, C. J. Fontes, and H. Zhang, *Los Alamos opacities: Transition from LEDCOP to ATOMIC*, AIP Conference Proceedings, 730 (2004), 168–179.

16. NASA, *Curiosity rover mission overview*, 2019. https://mars.nasa.gov/msl/mission/overview/.

17. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine learning in Python*, J. Mach. Learn. Res. 12 (2011), 2825–2830.

18. J. Salvatier, T. V. Wiecki, and C. Fonnesbeck, *Probabilistic programming in Python using PyMC3*, PeerJ Comput. Sci. 2 (2016), e55. https://doi.org/10.7717/peerj-cs.55.

19. D. J. Walters, A. Biswas, E. C. Lawrence, D. C. Francom, D. J. Luscher, D. A. Fredenburg, K. R. Moran, C. M. Sweeney, R. L. Sandberg, J. P. Ahrens, and C. A. Bolme, *Bayesian calibration of strength parameters using hydrocode simulations of symmetric impact shock experiments of al-5083*, J. Appl. Phys. 124(20) (2018), 205105. https://doi.org/10.1063/1.5051442.

**How to cite this article**: Hébert C-A, Lawrence E, Myers K, Colgan JP, Judge EJ. An initial exploration of Bayesian model calibration for estimating the composition of rocks and soils on Mars. *Stat Anal Data Min: The ASA Data Sci Journal*. 2021;14:596–605. https://doi.org/10.1002/sam.11503