Gaussian process emulators are very effective at making fast predictions of complex scientific computations that usually require very computationally expensive simulations (Higdon et al. 2008). After training (described below), the emulator is able to predict the simulator output at new, untried settings of the inputs, but in a fraction of the time. This speed increase makes the emulator useful for tasks like parameter estimation and model validation. In cosmology, this scheme makes predictions of the nonlinear matter power spectrum that match the results computed from brute force N-body simulations to a few percent (Lawrence et al. 2010). We will describe some of our preparation for the next generation of cosmological emulators.

**Emulation and the Original CosmicEmu**

We choose 37 settings of the inputs (left below) at which to run the expensive simulation. Each simulation produces a quantity of interest, usually functional (right below).

The functions are decomposed in a set of basis vectors (left below) and weights (right below). These are usually based on an SVD (i.e. principal components).

The 19 points fall on the intersection of a regular lattice (transformation of the 2-d integers) and the design region of interest (unit square). The right panel shows 37 runs that are the intersection of the region of interest and a dilation of the same lattice. This dilation results in a set of 37 runs that include the 19 runs from the left panel.

This process can be repeated, doubling the size of the design each time. At each stage, there is an approximately uniform distribution. Our design has three stages with complete lattices at 26, 55, and 101 runs (26 and 101 are each stage, there is an approximately uniform distribution. Our design has three stages with complete lattices at 26, 55, and 101 runs (26 and 101 are shown below). When the simulations for each stage are complete, we will release an emulator based on the results.

A smaller issue is that the current best fit value of the neutron mass is zero, on the boundary of its range. To improve prediction near this value, we supplement the simulation suite with 10 runs from a Latin hypercube over the other 7 parameters, with the neutron mass set to zero.

**Lattice Designs for Partial-Result Emulation**

The suite of simulations, the Mira Universe, for the next generation emulator will take a number of years to complete. In total, we will produce about 100 runs over an input space of 8 parameters. An emulator based on this expanded set of parameters will be a valuable addition to the astro community. Therefore, we would like to release new software without waiting for the entire suite to be ready. However, each release needs to be based on a set of inputs that fill the space uniformly. To solve this problem, we use an input design based on lattices (Berger 2011).

This design strategy is easily explained in the figure below. The left panel shows a 19-run design in two dimensions. These 19 points fall on the intersection of a regular lattice (transformation of the 2-d integers) and the design region of interest (unit square). The right panel shows 37 runs that are the intersection of the region of interest and a dilation of the same lattice. This dilation results in a set of 37 runs that include the 19 runs from the left panel.

In order to maintain stationarity but retain our modeling simplicity, we explore some transformations of our input parameters. The above results suggest a polynomial transformation of the sum of the parameters. In the end, the best results are achieved by transforming the original parameters, \( w \), and \( w_0 \) into \( w \) and \( (w-w_0)^{1.25} \). On this scale, our test results are considerably improved, as shown below.

**References**