Meaningful Automated Statistical Analysis of Large Computational Clusters*

J. M. Brandt, A. C. Gentile, Y. M. Marzouk, and P. P. Pébay
Sandia National Laboratories, Livermore, CA 94550, U.S.A.

Abstract

As clusters utilizing commercial off-the-shelf technology have grown from tens to thousands of nodes and typical job sizes have likewise increased, much effort has been devoted to improving the scalability of message-passing fabrics, schedulers, and storage. Largely ignored, however, has been the issue of predicting node failure, which also has a large impact on scalability. In fact, more than ten years into cluster computing, we are still managing this issue on a node-by-node basis even though available diagnostic data has grown immensely.

We have built a tool that uses the statistical similarity of the large number of nodes in a cluster to infer the health of each individual node. In the poster, we first present real data and statistical calculations as foundational material and justification for our claims of similarity. Next we present our methodology and its implications for early notification of deviation from normal behavior, problem diagnosis, automatic code restart via interaction with scheduler, and airflow distribution monitoring in the machine room. A framework addressing scalability is discussed briefly. Lastly, we present case studies showing how our methodology has been used to detect aberrant nodes whose deviations are still far below the detection level of traditional methods. A summary of the results of the case studies appears below.

Cluster setup Our cluster consists of 112 nodes mounted in 4 non-identical vertical racks. A node's air intake is in the front and exhaust is through the back (mesh doors exist on rack fronts and backs). Near the middle of each rack is a predominantly empty gap. Air enters the room through perforated floor tiles and exhausts through grates in the ceiling. Among other variables, temperature and fan speed are obtainable on a per processor basis. Temperature is reported in integer degrees. The fan has a few discrete states and is automatically raised in response to high CPU temperatures.

Job ensemble: discovery of an abnormal node Our first study was on a 4-node cluster subset. While its size limits the rigor of statistical conclusions, it does ensure relative uniformity of environment. This case also illustrates typical behavior on cluster nodes during a job. Figure 1, left, shows the temperature history of the four nodes in a test run. Nodes are initially idle. When a job is started the temperature increases rapidly, then eventually plateaus. When the job ends temperatures return to their idle values. Some variation exists in the values between nodes, but in general nodes in the same job group move in thermal concert with each other as typically they are doing the same types of tasks in parallel. We then deliberately altered one node, loosening the thermal bond between the processor and its heat pipe, causing it to dissipate heat less efficiently. Figure 1, right, shows the time history for the same test run in the altered case. When idle, the altered node was not meaningfully different from other nodes. However, under load, the altered node's thermal profile showed a significantly higher temperature rise than those of its peers. This behavior can be detected using simple statistics: for each node i, mean temperature $\overline{T}_i$ and standard deviation σ_{T_i} are computed on the other three (peer) nodes. On one hand, the temperature of the altered node A under load falls out of $[\overline{T}_A - 2\sigma_{T_A}, \overline{T}_A + 2\sigma_{T_A}]$, and thus the node is flagged as abnormal. On the other hand, the same analysis characterizes each of the other nodes as normal. Note that for this case the altered node is not detected at idle temperature. This example shows that (1) in a well-behaved environment, detection of the faulty node can occur at temperatures lower than traditional threshold values, and (2) some problems may only (or more strongly) manifest themselves under specific conditions, such as increased load. One must therefore be careful to consider ensembles under a range of conditions. In the remaining cases, then, we consider statistics with the added complication of a real-world environment.

Geographic ensembles: a fixable abnormal environment In this case we examined the average fan speed of nodes under production conditions in our cluster. Despite changing load in the cluster, the vast majority of nodes exhibited the same stable fan speed over time, with only a few nodes

*Extended abstract published in *Proceedings of the 2005 IEEE International Conference on Cluster Computing*, September 2005

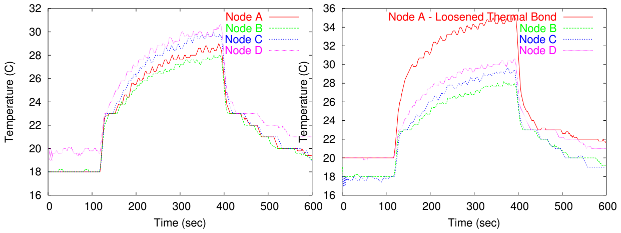

Figure 1.

exhibiting different behaviors. We considered several geographic ensembles, expecting the discordant values to be located either closest to the ceiling or closely co-located indicating a problem hot spot. Consideration of the average fan speed as a function of height showed that fan speeds in the rows bordering the gaps in the center of the racks were the highest. Examination showed that the basic physical cluster design was flawed as hot exhaust air from the back of the racks was recirculated through the gap in into the intake of the machines. Blocking this gap caused the errant fan speeds to fall in line with the values of the rest of the cluster. Note that this condition was not even hinted at by the traditional per-node threshold-based monitoring supplied with the cluster.

Correlating independent job ensembles: global effects We examined the correlation between the time averaged values of the temperatures of two different job groups. While typically the long term temperature behavior of a job group is relatively stable, in this case we found a time-frame in which the temperature of each job group rose and fell a few degrees in concert with each other (Fig. 2 left). Given the independence of the jobs this suggests that the change is due to some global factor. The size and time period of the change are consistent with a room temperature drift. Cross-correlation of jobs in this fashion allows one to grossly account for environmental conditions and to identify potential changes in the environmental configuration that can have large ramifications, such as the relocation of another piece of equipment that alters airflow.

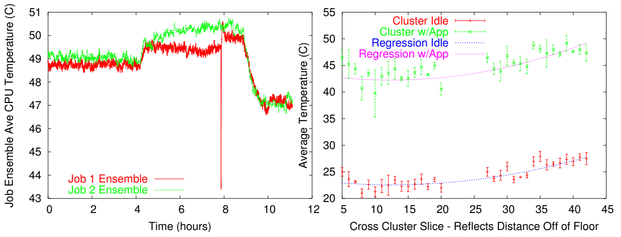

Figure 2.

Correlating job ensembles: avoiding false positives Some behaviors may mistakenly seem to be inexplicable and hence reflective of a problem. For instance Fig. 2, left, also shows a large temperature decrease at (at time around 8 hours) in job #1. Cross-correlation with job #2 indicates that this is not a global cluster effect. Ensemble examination showed this happens to all nodes in the job, which, from scheduler data are widely distributed throughout the cluster, making a nodal problem unlikely. We thus considered additional variables which might have secondary effects on temperature. Correlation with transmission data shows that the temperature decrease is not a problem, but a reflection of a normal phenomenon: computation stops while the job writes data to disk.

Geographic ensembles: unavoidable environmental field Fig. 2, right, shows the time-averaged temperature across the cluster as a function of distance from the floor, when all nodes are either (1) idle, or (2) running the same code. Near the floor, temperatures unexpectedly decrease with height. This decrease is due to warm air recirculating beneath the cabinet from the exhaust side back to the front because of a low pressure zone created by high velocity air emitted from the floor tiles. This skew from a completely homogeneous case surpasses the limit of what we can easily fix by tweaking the cluster design or making easy airflow alterations. We seek, then, to treat this skew as a baseline we can subtract so that it does not mask other normal variations in the system. We consider the extreme scenarios of the cluster at idle and under load, attempting to find models to fit the data for these cases. A natural approach consists of modeling temperature as some function of height, multiplied by some random "noise" that is on average equal to 1. The noise factor accounts for the parameters other than height that have an effect on temperature; there are evidently a great many of them, only considering manufacturing variations. A grasp at their relative importance with respect to height can be obtained by looking at the standard deviation of the noise – a standard deviation of zero would mean an exact fit to the model approximation. We fit the observed data to the model $T \sim \mathcal{N}(Q(h), \sigma)$ where h and T respectively denote height and temperature, $\mathcal{N}(Q(h), \sigma)$ is the normal distribution with mean $Q(h)$ and variance σ, and Q is a quadratic polynomial whose coefficients need be identified, as well as σ. We were able to fit each of the idle and running data with a confidence level of up to 99%. Variance is greater for the computationally intensive case which seems to emphasize the physical discrepancies between computational nodes. Figure 2, right shows data and models for both the idle and the computationally intensive case. We thus establish that the modeled skew can be subtracted.

Finally we note that the ability to categorize the natural environment is valuable information in and of itself. Nodes running at consistently higher temperatures due to the environmental field may be susceptible to more problems, decreased performance, and shorter lifespan than those located elsewhere.

[1] J. M. Brandt, A. C. Gentile, Y. M. Marzouk, and P. P. Pébay. Meaningful statistical analysis of large computational clusters. Sandia Report SAND2005-4558, Sandia National Laboratories, July 2005.