Timeseries Interferometry in Distributed Systems (proposal)

Mark Burgess
Oslo University College
mark.burgess@iu.hio.no

***Abstract*—An approach to simplified correlation of process anomalies is proposed, inspired by the method of interferometry of signals in optics. This method could form the basis for one or more Master of Science research projects.**

This proposal 4 November 2006 for academic year 2007. Contact Professor Mark Burgess (mark.burgess@iu.hio.no) if you are interested in this project.

I. Introduction

In [1], [2], it was shown how the running state of a computer process, or collection of processes, could be classified using a simple form of machine learning to quickly build estimators of average values with small deviations in observables. These values could then be translated into symbolic values with simple baseline semantics based on relative state. The approach was aimed at autonomous systems performing self-diagnosis (introspection), and allowed end users to determine the final semantics of complex states through policy. A proof of concept has been implemented in Cfengine, with around 5 years of experience.

This work was later developed to consider the possibility of more sophisticated events and correlations [3], [4] between time-series belonging to the same process or host, and between multiple processes and/or hosts. Typically, the semantics of distributed state are harder to define and form policy about. The success of leap detection has yet to be proven in a convincing way.

In many cases, monitoring's purpose is to know when to alert changes (either to a human operator or to an autonomic error correction engine like Cfengine). Simple adaptive thresholds based on de-trended statistics can be fairly reliable on busy hosts, where there sufficient samples to form a regular sample, but quite unreliable of little-used systems. The latter seems unavoidable; a lack of information cannot be repaired without more information. In the former case, a few improvements for sudden change detection in the temporal patterns were also considered with leap detection analysis of correlations [3].

In this paper, the aim is to study what methods might be used to provide a calibrated, centralized view of this approach to timeseries, something like a central 'brain' correlating multiple sensors. The goal of this would be to determine whether any inferences can be made about distributed state, without massive offline data collection.

As in the earlier studies, the purpose here is not to enable forensic analysis of historical data, but rather to quickly model actual running state for quick adaptive response.

II. Centralization of compressed timeseries data

Each monitoring agent[1] samples data into discrete time $\Delta\tau$ buckets, every $\Delta\tau/2$ seconds, thus forming a periodic clock, modulo the working weekly trend (thus we are limited to measuring changes of order $\Delta\tau$ and greater).

Each distributed agent collects tuples of data of the form $(q(t), \langle q\rangle(\tau), \sigma_q(\tau), H(\tau), H)$, where $q(t)$ is the variable being sampled, t is the actual time measured on the process clock, $\langle q\rangle(\tau)$ is the estimated running mean value at periodic time τ, $\sigma(\tau)$ is the estimated standard deviation, and the histogram of all values , with $H(\tau)$ and without H detrending [5].

The value of aggregating signals for a comparative analysis will surely depend on a few things:

- The cost of aggregation of data to central store. The amount of data transmitted must be small.
- The cost of comparison of time-series across all sources. The computational cost must be small.
- How quickly we forget data is also an important issue in remaining sensitive to anomalies. The relevance of data needs to be kept fresh.

We aim to minimize the cost of collection and computation. Just as the geometric 'Bayesian' average used in the two-dimensional time parameterization reduced the compuation of running mean and standard deviation to an O(1) calculation, of the convex form:

$$q'' \rightarrow \frac{\alpha q + \beta q'}{\alpha + \beta} \tag{1}$$

By the same token, we wish to reduce the computational cost of correlation to a minimum too in cross sections. This can be done by redigitization and logical AND, OR, XOR operations.

III. State compression

Since the end agents perform all the computation in a distributed manner, and compress all the threshold identification and state classification into just a few symbols, one can now compress these symbols into a tiny bit encoding for transport to a central hub or 'brain'. For example,

Based on the learned scale estimators for $\langle q\rangle(\tau)$ and $\sigma(\tau)$, one can take a process (e.g. incoming WWW connections), and classify the current context into a number of symbolic states:

[1] In cfengine, the agent was called cfenvd in cfengine 2, and will be called cf-monitord in cfengine 3.

```
www_in_high_dev1
www_in_high_dev2
www_in_high_anomaly
...
www_in_low_anomaly
```

For example, one possible encoding (per process, here 'www in') might be:

sgn	σ	2σ	$\geq 3\sigma$

Thus a small 2 standard deviation positive spike would be encoded as 0110. A negative drop by one standard deviation would be 1100, etc. The final coding will be informed by experiment.

Some tuning of these bit positions will be needed. For instance, a single standard deviation is unlikely to have a sufficiently low statistical uncertainty to draw a conlusion, except perhaps in the small number of metrics that are basically constant or Gaussian (user driven processes). We can take this based on the idea that small variations are simply noise and have no significance, thus it is only larger units of change measured in units of $\sigma(\tau)$, at each time bucket τ.

By packing symbolic compressed state (with fixed semantics) into a single long bit-string (e.g. 64 bit words) in a preconsidered way, it becomes very cheap to collect data and make comparisons. A correlation analysis then becomes simple a logical AND. The speed of this analysis begins to make realtime reasoning about global state plausible, without the kind of offline clusters used in intrusion detection.

By placing σ and 2σ as neighbouring bits (rather than using a Gödel numbering coding) we can blur the lines between σ and 2σ when correlating, just be ORing bits together.

A correlation now becomes a simple AND bit operation with no arithemetic steps, and all data samples have been compressed into just 4 bits. The sign bit is used to say whether the current deviation is above or below the running mean estimator. Anti-correlations can then be performed by XORing the sign bit.

IV. Principal Component ellipsoid

The ellipsoidal distribtion of correlations between signals in a multi-source signal gets reduced to a digitized oblong hypercube in the current method. Again, the summarization and re-digitization of state by applying the anomaly classification makes PCA extremely cheap too. This might be useful in characterizing the major sources of (un)certainty in a process group, giving clues about possible channels of common causal orgin.

V. The problem of multi-modal and long-tailed distributions

The $\sigma(\tau)$ estimator (first moment of the τ slice distribution) is a reasonable adaptive scale for measuring significance, but it is limited to mono-modal distributions without long tails. The value of an automated anomaly detection thus comes from analysing the various histogram view points (with and without de-trending) to identify hidden scale-pinnings and boundary conditions. Long tailed distributions tend to be time-related simply due to the finite size of process resources, and scale-free phenomena that lead to power laws are also rare in system metrics. This remains to be seen, however. In cf-monitord, the Hurst exponent has been used to estimate the likelihood of self-similarity. While theoretically attractive, this yields many false positives.

VI. Path interferometry

In astronomy, geology and material science, interference patterns are used to detect changes using multiple paths to the same point. This is not the usage I am advocating here.

Transit times are difficult to measure as they require a round-trip time. Since distributed agents each have their own clock (which cannot be assumed in synchrony) one cannot expect to relay on start times measured in more than one location.

VII. Forensics

The method proposed here is intended for real-time state adaptation (self-healing, computer immunology), but the data might also have long term value.

The problem of keeping raw data (without semantic context) over long times, as advocated by many monitoring systems, seems pointless as the interpretation of data many years after the fact is decreasingly likely to be sensible (not that this stops anyone from doing it(!)). Applying naive statistics to data without a context breaks the basic rules of a scientific understanding.

However, suppose we store this encoding over long times. Now the semantics are completely defined by the model, and have symbolic meaning. This meaning is invariant, even as the states change, and thus we have a kind of semantic coordinate system for the data.

Investigating the latter possibility will probably take longer than the project duration, but could be the basis of future work.

References

[1] M. Burgess. Two dimensional time-series for anomaly detection and regulation in adaptive systems. *Lecture Notes in Computer Science, IFIP/IEEE 13th International Workshop on Distributed Systems: Operations and Management (DSOM 2002)*, 2506:169, 2002.

[2] M. Burgess. Probabilistic anomaly detection in distributed computer networks. *Science of Computer Programming*, 60(1):1–26, 2006.

[3] K. Begnum and M. Burgess. Principle components and importance ranking of distributed anomalies. *Machine Learning Journal*, 58:217–230, 2005.

[4] K. Begnum and M. Burgess. Improving anomaly detection event analysis using the eventrank algorithm. *Lecture Notes on Computer Science*, 4543 (Proceedings of the first International Conference on Autonomous Infrastructure and Security (AIMS)):143–154, 2007.

[5] M. Burgess. The kinematics of distributed computer transactions. *International Journal of Modern Physics*, **C**12:759–789, 2000.