ADAPTIVE RESOURCE PROFILING

Dennis DeCoste
Machine Learning Systems Group
Jet Propulsion Laboratory / California Institute of Technology
4800 Oak Grove Drive; Pasadena, CA, 91109; USA
http://www-aig.jpl.nasa.gov/home/decoste/
decoste@aig.jpl.nasa.gov

Abstract

This paper addresses the problem of learning *resource profiles*: upper and lower prediction bounds for engineering resources (e.g. power). We argue for data-driven techniques which specifically learn interval-valued predictions (i.e. best and worst-case bounding functions), as opposed to probabilistic (i.e. soft) predictions. We present and discuss a simple preliminary example using actual data from a rover prototype in a laboratory test-bed environment. We discuss extensions to this work, including integration with an onboard planner that could use these resource profiles, toward improving overall science throughput.

1 Introduction

This work addresses the problem of automated learning and updating of resource models (e.g. battery power availability) using sensor data. Robust spacecraft/rover autonomy requires the ability to maintain resource models onboard, to reflect changing environmental and degrading spacecraft conditions with minimal ground attention. Furthermore, future planned missions and economic constraints present increasing pressures to deal with largely unknown environments and short design-build-launch cycles (with minimal time for rigorous testing). Thus, careful pre-flight manual preparation of resource models is likely to be infeasible and inadequate.

The traditional ground-based approach (send all data to ground, perform trending and statistical modeling manually, update models) is both suboptimal and impractical. It results in reduced science throughput, due to both spacecraft-ground communication delays and the need to use excessively conservative resource margins. Furthermore, for key future mission contexts such as multiple cooperating rovers, spacecraft fleets, and Deeper Space missions (such as planned Pluto flybys), the telemetry bandwidth requirements and/or communication delays would be enormous.

1.1 Adaptive Resource Profiling

To address such problems, we have developed machine learning and data mining techniques to both learn initial resource models from historic sensor data (e.g. testbeds, simulations, early mission behavior) and to continually adapt them using online sensor data. Specifically, we have adapted our earlier work in abnormality/fault detection via learning red-line envelope functions ([1,2]) to the task of *resource profiling*: learning upper and lower bounds on expected future resource availability over time.

Each profile projects how much of a resource may be available over future time points, based on the current resource level and on the durations of actions which can produce (e.g. activate solar panel) or consume (e.g. turn on motor) the target resource (e.g. battery power). To reflect uncertainty in the impact of such actions (due to both unobserved-yet-contextually-significant effects and routine sensing noise), these predictions can be based on learned context-sensitive interval-valued (rather than nominal mean-valued) estimates of the production and consumption rates of such actions.

The end result is an envelope profile showing the best-case and worst-case resource availability over time. Such profiles are useful both for plan execution monitoring (i.e. when actions are observed) and planning/scheduling (when future actions are planned). These models allow reasoning under both best and worst-case scenarios, to guide aggressive attempts toward maximum science throughput while avoiding controlling dangerously close to worst-case limits (e.g. heading into night-fall without sufficient battery charge to run critical night-time operations or experiments).

Details for some techniques to learn bounding func-

Proc. Fifth International Symposium on Artificial Intelligence, Robotics and Automation in Space, 1-3 June 1999 (ESA SP-440)

tions from data can be found in ([1],[2]). The key property of these techniques is that they: 1) result in few false alarms (e.g. they properly contain all the data within the resulting bounding intervals) while otherwise striving to be as tight as possible and 2) overcome key limitations of other data-driven alternatives. For example, the common approach of *error bars* (e.g. neural network predictions of means and variances) make strong assumptions about the nature of the prediction error distribution (e.g. symmetric Gaussian noise). More general non-parametric ***probably density estimation*** overcomes that problem, but tends to be very "data hungry" and spends significant effort modelling the nature of the data between the extrema values.

In contrast, our bounding techniques essentially view the problem as a form of constrained optimization: make predictions which are as close to the target (e.g. in the least squared error sense) while ensuring that those predictions are always above (or below, in the case of low bounds) the target values. Our technique does not spend effort modelling the entire probability spread (only predicting the context-sensitive extrema values), nor does it make strong assumptions about the nature of the prediction error distribution, except that the maximum error is bounded (i.e. finite range between the tails of the distribution), as is typically the case in practice for digitally-sampled engineering data.

2 Example: Mars Rover Battery Drain

As a concrete example, consider a key resource for a Mars rover: power. Solar panels provide power, loads (e.g. motors, cameras) consume it, and the battery stores it. There is uncertainty in the rate at which the solar panels recharge the battery, due to conditions such as dust accumulation and Sun position. The loads also have uncertain consumption, due to variabilities that existing sensors are inadequate to capture. In our experiments (running the Rocky 7 rover prototype in the JPL Mars Yard), the possible training inputs are: 1) sensed quantities such as battery voltage, wheel motor torques and currents, and solar intensity and 2) the times of various actions (such as turning on/off cameras, which do not have their own sensors of currents).

From actual sensed data of such inputs over time, our system learns to predict bounds on the battery power at any given time T+deltaT into the future. In our experiments with Rocky 7 so far, these predictions are based on the current battery voltage (and other selected sensors) at time T, as well as some fixed lagged time values in the past (e.g. at T-L1, T-L2, ...). In practice, a prediction target of the remaining kilo-watt/hours of power (instead of voltage levels per se) is more meaningful. This requires computing backwards from a final (0 KW/hr) battery dead state, computation of load watt requirements (i.e. from observed current and duration data), and integrating to compute target values of "remaining power" over each sensed time point. For simplicity, the experiments discussed below focus on predicting the voltage level.

2.1 Example Performance

Figure 1 shows the training data, consisting of 23 sensors over about 7.5 hours. This data was gathered over six independent trials of Rocky 7, under various load and solar conditions, and combined into one time-series data-set. [1] Each trial was run from a full battery charge until the battery power dropped so low that the CPU and data sampling shut down. The solar panel on the Rocky 7 prototype is actually insufficient to recharge the battery, even with no loads other than CPU; so, it merely slows down the power drain rate. Thus, the plot of the battery voltage sensor (labelled **MezVoltage-batteries**) shows 6 distinct periods of high-to-low voltage drop, one for each trial.

Figure 2 shows the same sensors, for the single test (seventh) trial. The test trial was about one third the duration of a nominal (no load) battery drain trial, due to especially heavy loads (i.e. much wheel motor activity). Figure 3 shows the evaluation of the learned battery resource profiles when applied to the test data. Those high and low resource bounds were learned using only the training data, for a prediction forward lag of 1 minute (i.e. deltaT=60). The inputs for this example were various lagged values of the battery voltage (specifically, T,T-1,T-2,T-4, and T-8). The test data completely fits within the bounds. The noticable looseness is a result of having learned bounds which contain all 6 of the training trials; a profile learned for this test trial alone would be much tighter (but more prone to not fit future data). The looseness is especially obvious for the high bound. This arises from the fact that the training trials involved a variety of loads, some much less than for the test trial. Including relevant action events (such as motors being on or off) as inputs to these bounding functions would lead to tighter predictions.

[1] Data gap periods of 1000 seconds (not shown in plots) were inserted between each of the six training data subsets, to avoid lag vectors from crossing any trial boundaries.

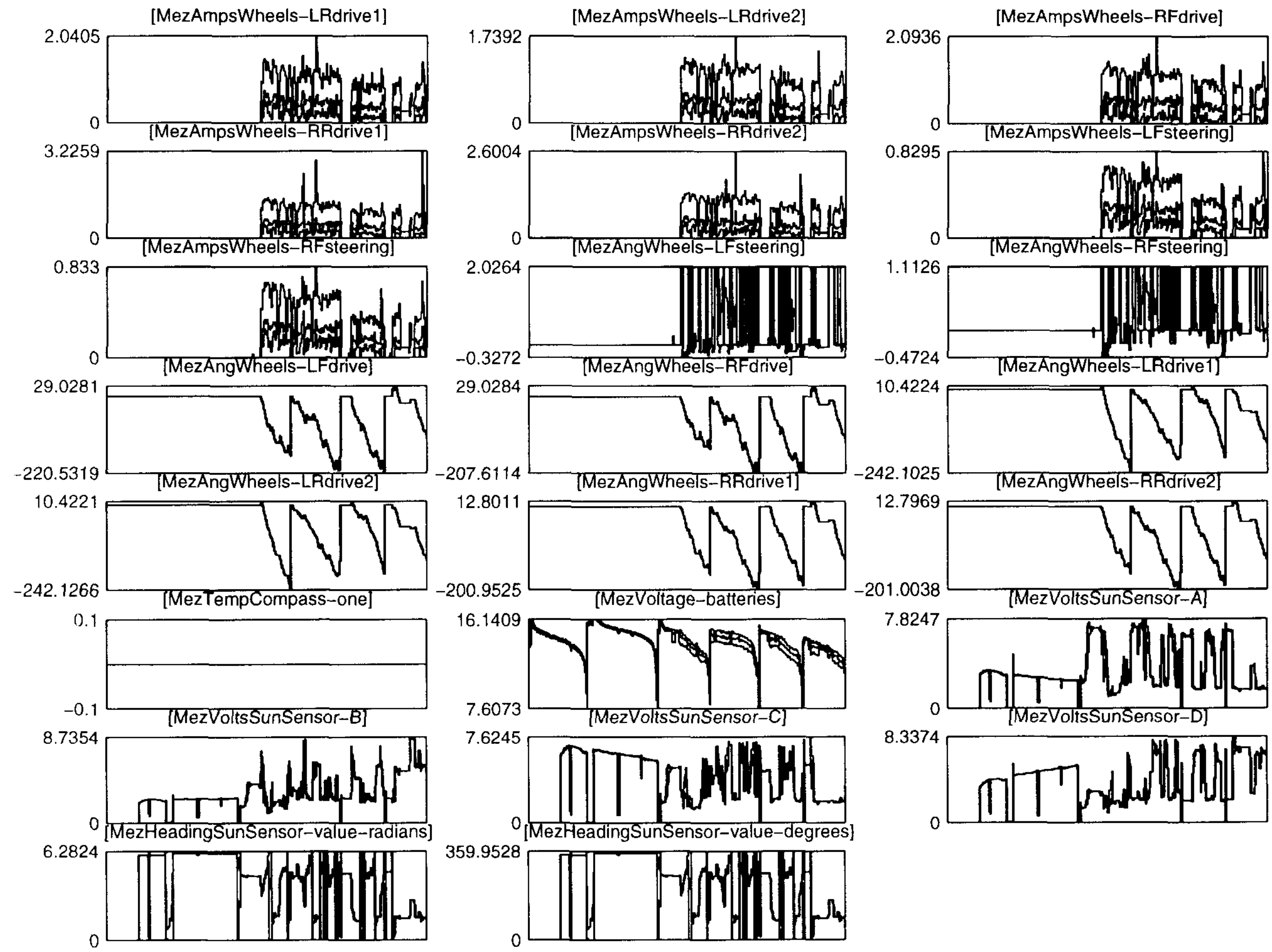

Figure 1: **Plot of training data**

Each box is a time-series plot of the named sensor. These time-series represent a concatenated sequence of six test-bed experiments, as suggested by the battery voltage (plot box which is 6th down, 2nd across) starting six times at high (full) values (near 16 volts).

3 Discussion

For more accurate, context-sensitive resource profiling, the inputs to the bounding functions should include quantities related to the actual loads over time. However, simply using the relevant raw sensor data (e.g. motor currents) will generally not work well. For resources such as battery power, integration over time windows greater than a few local samples is often effectively required, to model with sufficient precision the contributions and depletions of the underlying resource quantity (e.g. power).

Thus, we are investigating using features representing the total duration of various actions (e.g. camera on, motors on) between the current time T and the predicted time T+deltaT. We believe that using such aggregate durations for each type of load activity (e.g. number of seconds motor 1 is on between time T and T+deltaT) as inputs, instead of the sensed quantities of those loads (e.g. actual electric current values at each motor over time) per se, also provide more useful models for use in resource management by planners. This is because a planner will reason at the level of such actions, and our model must itself be able to map those actions into worst-case and best-case consumption rates. Our use of such load-activity durations as inputs does reflect an assumption that the resource consumption is an additive function of such durations. Balancing the predictive imprecision that results from such abstractions, while still providing useful abstracted interfaces for planners (i.e. not at the detailed level of sensed load quantities), is our

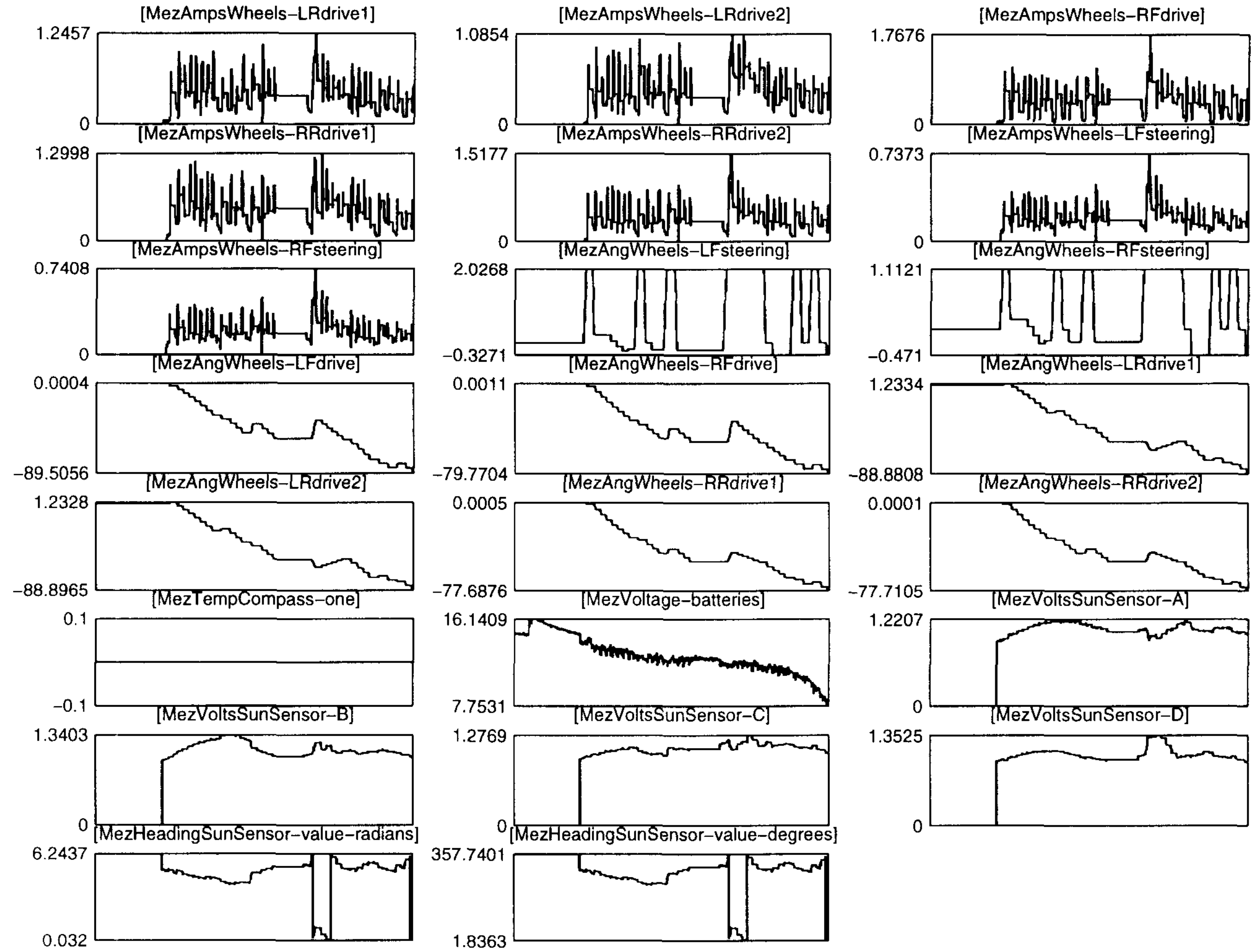

Figure 2: **Plot of test data**

current focus in ongoing research.

To overcome the expense and limited sample size of current testbed testing, we are currently evaluating these techniques on simulated Rover data, under a variety of load and action contexts. We plan to more tightly integrate this resource profiling capability with existing automated planning capabilities over the coming months (for preliminary architecture for such integration, see [3]).

4 Acknowledgements

Eddie Tunstel and John Szijjarto provided significant feedback and assistance on this work.

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

5 References

[1] Dennis DeCoste. "Mining multivariate time-series sensor data to discover behavior envelopes". Proceedings of the Third Conference on Knowledge Discovery and Data Mining (KDD-97), Newport Beach, CA, August 1997.

[2] Dennis DeCoste. "Automated learning and monitoring of limit functions". Proceedings of the Fourth International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS-97), Japan, July 1997.

[3] Tara Estlin et al. "An Integrated Architecture for Co-operating Rovers". Proceedings of the Fifth International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS-99), The Netherlands, June 1999.

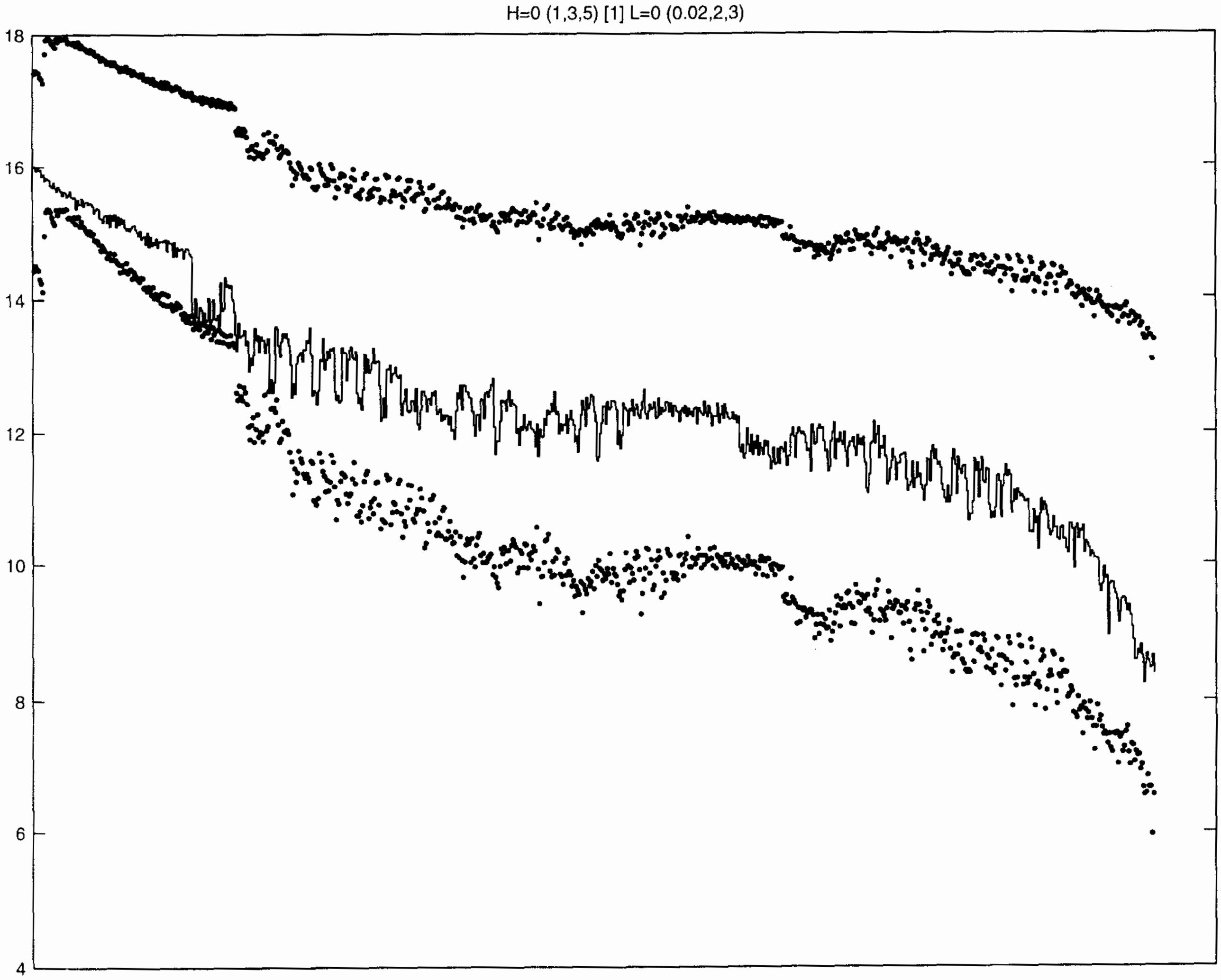

Figure 3: Plot of battery envelopes for test data

Time-series plot of upper and lower profile bound values, for 1 minute look-ahead prediction. The actual test data is between these bounds over all time points.