Assessment of Overhaul Effectiveness and Usage-based Inference using Bayesian Networks

Nenad G. Nenadic[1], Christopher J. Valant[2], Sean P. McConky[3], and Michael G. Thurston[4]

[1,2,3,4] *Rochester Institute of Technology, Rochester, NY, 14623, USA*
nxnasp@rit.edu, cxvgis@rit.edu, spm9605@rit.edu, mgtasp@rit.edu

Abstract

The process of assessment of effectiveness of the existing overhaul practices determined that the historical usage of assets provides valuable contextual information. Usage data is typically highly reliable, but not in legacy fleets, featuring older vehicles with missing, incomplete, inconsistent, and contradictory data. This paper describes two methods for usage estimation from noisy data by exploiting two data sources: 1) unreliable, manually-entered usage data and 2) part replacements. The first method employs a probabilistic model to reconcile missing and inconsistent data entries; the second is based on the replacement of consumable components. The probabilistic model, fully and uniquely specified by the probabilistic variables (with their distributions) and deterministic variables, is validated using synthetic datasets because the real ground truth associated with the field data does not exist. Disproportional impact of an incorrect initial data point is mitigated by training the model in both forward and reverse directions. The motivating hypothesis for usage estimation from part replacements is based on a plausible assumption that specific consumables, e.g. brake pads, have reasonably repeatable replacement patterns which can be related to usage. For many vehicles mean time between failures of a component was even longer than the average data collection time span. But for assets with sufficiently longer data records, the cumulative replacements of components are well-correlated with the probabilistic usage estimates, providing additional reinforcement for the inference.

1. Background

Equipment health and condition monitoring enables maintenance to better manage the effects of equipment degradation or failure. The systematic approaches include Reliability Centered Maintenance (RCM) and Condition-Based Maintenance Plus (CBM). RCM specifies processes that must be implemented to manage failure modes that can cause functional failure of assets, i.e. RCM identifies the maintenance needs for CBM (Nowlan & Heap, 1978; Moubray, 1997). CBM is the application and integration of selected processes, technologies, and knowledge-based capabilities to improve the reliability and maintenance effectiveness of systems and components (US-AMRDEC, 2016). Being based upon objective evidence of equipment degradation or impending failure, CBM has significant economic and safety benefits; it reduces incidence of unscheduled failures and downtime, and the occurrence of unnecessary or early scheduled maintenance.

Health or condition monitoring is the process of collecting asset data and extracting the information for CBM. Affordable sensors, data storage, and networking enable comprehensive monitoring of all types of assets. In order to make this data actionable for CBM, specific models are necessary to identify and characterize anomalies to relate the anomalous patterns to forward looking failure risk for decision making purposes (Engel, Gilmartin, Bongort, & Hess, 2000; Goebel et al., 2017). The models are typically classified as expert-system, physics-based, data-driven, and hybrid (Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006).

Health monitoring is often an incremental process, as data is typically not available to develop comprehensive diagnostic and prognostic algorithms from the outset. Instead, the levels of Prognostic Health Management (PHM) capability grow over time (Sikorska, Hodkiewicz, & Ma, 2011; Bussey, Nenadic, Ardis, & Thurston, 2014). Consolidation of vehicle fleet data in a data warehouse provides an opportunity to develop PHM knowledge and algorithms incrementally. The current study stems from an initiative to reevaluate the effectiveness of some of the existing maintenance practices. A specific objective of high importance was to assess the effectiveness of periodic overhauls based on existing data. The data came from a family of military tracked vehicles. The approach was pragmatic, based upon the available data, and available domain knowledge.

Knowledge of assets' usage has the potential to improve the assessment. However, because the usage data was unreliable (contained inconsistent and often missing values), an inference model was needed. A probabilistic networks model was selected because networks of Bayesian probability calculations are an effective way to impute missing data and improve statistical models of time-series data. Multiple imputation is an effective way to fill missing data while maintaining a degree of uncertainty (Honaker & King, 2010). Parameter estimation in statistical models can be used to build a belief network for correcting imperfect observations in data mining problems (Denoeux, 2013). Probability was first interpreted as a natural extension to formal logic in the 1940s (Cox, 1946, 1961). This approach was later extended and popularized by Jaynes (Jaynes, 2003) and others ((Sivia, 1996; Gregory, 2005).

The problem with probabilistic models was their computational requirements. The *Markov chain Monte Carlo* (MCMC) method, that also originated in the 1940s (Metropolis & Ulam, 1949), has become a standard tool for this computation. Probabilistic (graphical) models had notable successes in the sixties and seventies, but then fell out of favor to some other AI approaches (e.g. fuzzy logic (Zadeh, 1965, 1983), Dempster-Shafer evidence theory (Shafer et al., 1976)). The renaissance of probabilistic modelling started in the late 1980s with theoretical development (Pearl, 1988), but computational power did not allow the explosion of solutions that we have witnessed in the very recent past. The graphical approaches will allow domain experts to take advantage of the framework. Many new applications including medical diagnostics, analysis of genetic and genomic data, speech recognition, natural language processing, analysis of market data, and fault diagnosis, which can be extended to reliability. The potential of the probability networks has been recognized in the 1980s (Pearl, 1987; Geman & Geman, 1984)). More recently excellent tutorials (e.g. (Andrieu, De Freitas, Doucet, & Jordan, 2003; Diaconis, 2009) and books (e.g. (Jordan, 1998), (Bishop, 2006), (Barber, 2012), (Koller & Friedman, 2009), (Gelman, Carlin, Stern, & Rubin, 2014), and (Theodoridis, 2015)) have become available.

Probabilistic modeling is computationally intensive. Many high-quality, well-tested frameworks for probabilistic modeling emerged, including BUGS (Lunn, Thomas, Best, & Spiegelhalter, 2000), JAGS (Plummer et al., 2003), Church (Goodman, Mansinghka, Roy, Bonawitz, & Tarlow, 2012), Infer.NET (Minka, Winn, Guiver, & Knowles, 2012), PyMC (Patil, Huard, & Fonnesbeck, 2010; Davidson-Pilon, 2015), and Stan (Gelman, Lee, & Guo, 2015). Usage estimation in this work employed PyMC and Stan because of their ability to integrate with the open-source scientific Python ecosystem, which also includes NumPy (Oliphant, 2006), SciPy (Jones, Oliphant, Peterson, et al., n.d.), Matplotlib (Hunter, 2007), with statistical Pandas (McKinney, 2012), and machine learning tools, Scikit-Learn (Pedregosa et al., 2011)), Theano (Bergstra et al., 2010), TensorFlow, and PyTorch.

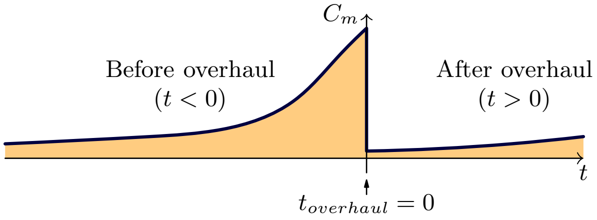

Figure 1. Sketch of expected maintenance cost before and after an overhaul assuming that the overhaul is effective.

2. INTRODUCTION

2.1. Approach based on the available data

For the fleet of interest, the maintenance history data was available, but the operational history was almost entirely lacking. While the knowledge of the operational history had the potential to greatly improve the decisions, the decisions had to be made even in the absence of important data.

A way to demonstrate effectiveness of previous vehicle overhaul using maintenance data alone was to visualize the cost of maintenance over time. Figure 1 shows a hypothesized maintenance cost C_m as a function of time t, where cost of maintenance was computed as the sum of the cost of parts C_{parts} and the cost of labor C_{labor}

$$C_m = C_{parts} + C_{labor} \tag{1}$$

where the cost of labor C_{labor} is a product of time associated with active repair t_{labor} expressed in hours (labor hours) and hourly rate r_{labor}

$$C_{labor} = r_{labor} t_{labor} \tag{2}$$

The tacit assumption in this figure was that the usage and the general operating and environmental conditions did not change appreciably over the time associated with an overhaul cycle. In practice, this assumption had to be verified, as shown in Section 3. Within the existing database, parts cost was generally available. However, the number of parts replaced were sometimes misleading and the recorded labor hours were often missing, requiring hand cleaning of data for analysis.

2.2. Maintenance Application Overhaul

The motivation of this study was in reducing the cost of overhauls by choosing more appropriate, condition-based maintenance times across the fleet. Figure 2 shows maintenance and overhaul activity on a fleet of more than four hundred vehicles. The solid (dark blue) line shows overhaul dates, the faded orange background signifies the time interval with recorded maintenance history corresponding to the overhauled vehicle and is populated with dark-orange dots that denote maintenance events, and the dashed grey line marks six

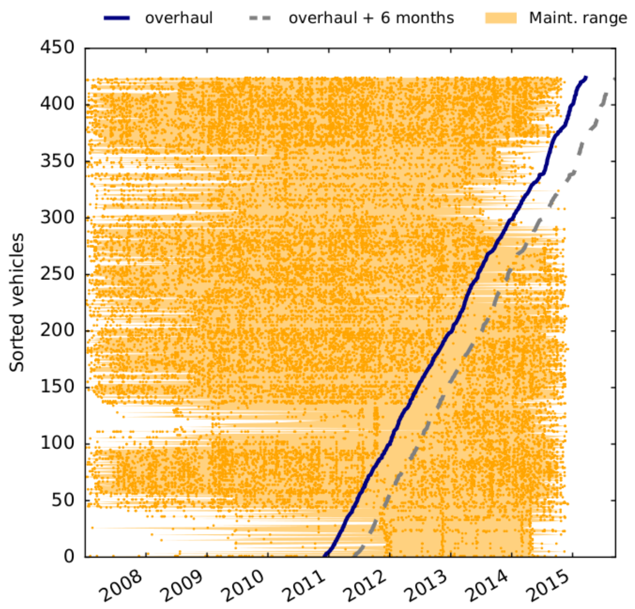

Figure 2. Overhaul (dark line) and maintenance events over time for a fleet of vehicles.

months after the overhauls. Note that there are generally few maintenance events in that six months interval after overhaul.

In order to assess the effectiveness of current overhauls, data was arranged so that all vehicle's overhaul dates aligned and cost records before and after could be compared. The purpose of this visualization is to compare cost, availability, and number of repairs relative to overhaul, across all vehicles. Cost and maintenance histories were analyzed by sub-system (i.e a collection of components categorized by their collective purpose). Components that were hypothesized to have replacement rates most influenced by usage, such as road wheels, were analyzed individually (Section 3.2).

A cost analysis was done for the 425 vehicles that have been overhauled since 2007 and data was grouped by 90 day intervals, as shown in Figure 3. From top to bottom, the figure shows a) total cost, b) available number of vehicles in the sample population, c) cost per available number of vehicles, and d) number of maintenance orders with non-zero cost. Grey and orange shading were used to emphasize the comparison of activity before and after the overhaul dateline. In order to avoid effects from the volatility of price variation over time, all costs shown in this plot were based on the 2014 unit prices using information on current parts. All vehicles in the sample contained the overhaul event. However there were fewer vehicles that had observation time intervals long before and long after the overhaul event. To take into the account the number of vehicles available at a time away from the overhaul event, the cost per vehicle is shown in Figure 3c. Figure 3b shows the availability of vehicles.

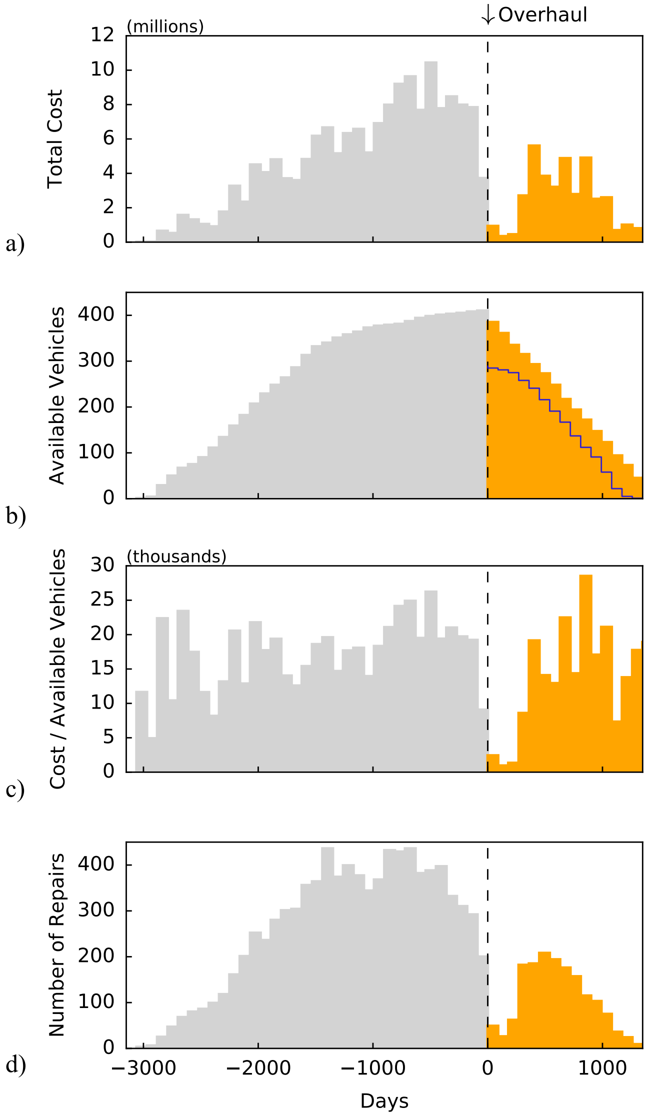

Figure 3. Effect of overhaul in time measured in days: a) total cost, b) available vehicles, c) normalized cost (per vehicle), and d) number of repairs.

The available number of vehicles was computed by taking the earliest and latest maintenance date for each vehicle and counting the number of vehicles available in the days relative to overhauls. The earliest maintenance dates were used to count number of available vehicles before overhaul and are represented by the grey shade. The latest maintenance dates were used to count available vehicles after overhaul and were represented initially by the blue line but later revised to the orange shade for the following reason. 132 vehicles did not have maintenance records after their overhaul, hence the gap in Figure 3b between the end of the grey shade before overhaul and the start of the blue line after overhaul. The lack of vehicle data after overhaul was also apparent in the top right region of Figure 2, where orange data points were sparse. For

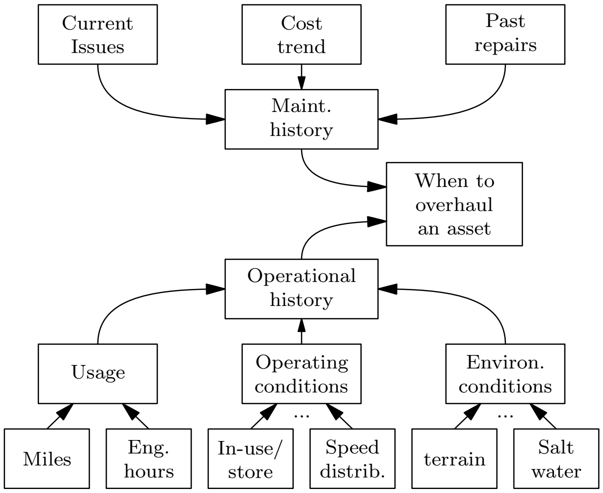

Figure 4. Data for making decisions on vehicle overhaul.

Figure 3b, the count was revised from the difference between overhauls and the latest maintenance events to the difference between overhauls and the time of analysis, since some vehicles had no maintenance events after their overhaul date and before this study.

Figure 3a and Figure 3c shows reduced maintenance cost for period of time after overhaul. However, as the maintenance moves from time-based scheduling, it needs to consider asset usage. The remainder of the document is concerned with the usage estimation.

2.3. Best case scenario

The objective of this study was to obtain inferential estimates of usage based on noisy raw data. This study supports a broader goal to improve the overhaul interval utilizing condition and usage information, which requires accurate usage estimates. Clearly, maintenance needs depend on the context, defined by operating and environmental conditions a vehicle is subjected to. Thus, a careful analysis of vehicle overhaul effectiveness for a given vehicle must consider the context of use; however, this type of data was not recorded. Instead, the consideration of vehicles collectively has the potential to reveal the impact of overhaul on the continuous vehicle maintenance.

Figure 4 shows factors that contribute to the state of health of a vehicle. Two main classes of influence are identified as the operational history and the maintenance history.

Operational history is determined by usage, operating conditions, and environmental conditions. The usage is typically measured in miles, engine hours, or both. Operating conditions include in-use/storage patterns, braking, sudden accelerations, engine and vehicle speed distributions, and other similar parameters. Environmental conditions include terrain, humidity, exposure to salt water, dust, etc. The dominant approach for capturing usage data is the inclusion of *health and usage monitoring system* (HUMS), defined as a system of sensors, processes, and algorithms for prognostics, on the vehicle platform (Heine & Barker, 2007). HUMS applications have been deployed only on the most expensive vehicles, such as fix-wing aircraft (Trammel, Vossler, & Feldmann, 1997) and rotor craft (Gordon, 1991; Ellerbrock, Shanthakumaran, & Halmos, 1999). Ground vehicles are generally not equipped with HUMS because HUMS adds on cost and complexity (with potential decrease of reliability), but it has been explored in research and development e.g. (Heine & Barker, 2007; Rabeno & Bounds, 2009; Das, Hall, Patel, McNamara, & Todd, 2012). The older tracked vehicles considered in this study were not equipped with HUMS. Maintenance history includes current issues, past repairs, and cost of maintenance over time.

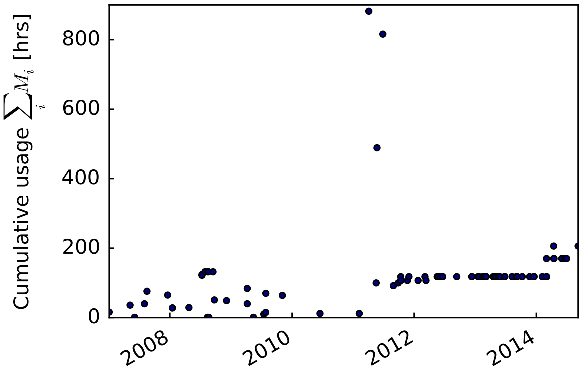

Figure 5. Usage data example: the values are not monotonically non-decreasing as expected

3. USAGE ESTIMATION

Of the operational history data described in Figure 4, only usage data existed in the available data set. The maintenance history database contained a field called *usage*, and for the fleet studies this value is supposed to represent cumulative engine hours (as read from the engine hour meter) or total miles. The units of recorded usage were unclear at first. Due to the noisy nature of the data, they may have represented hours driven, or miles traveled. Either unit should have non-decreasing values over time, but the volatility of the observations made it difficult to determine which were most plausible. Furthermore, observations were far enough apart in time that logical reasoning was not enough to distinguish the units (e.g. usage changing by five in one hour would imply miles, not hours). Further investigation of ordinance vehicle monthly and daily log-books strongly suggested the units of usage were hours. Of the 32 vehicle log-books provided to us, several held records that precisely matched that of the database for the vehicle under units of hours. However, usage data had many missing, inconsistent, and contradictory values. An illustration of this situation is shown in Figure 5,

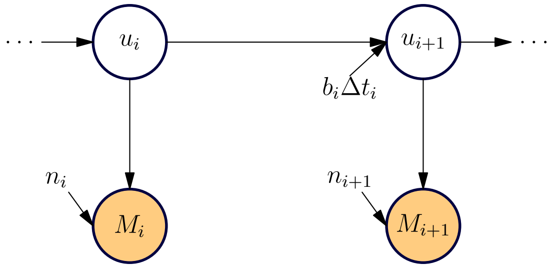

Figure 6. A simple probabilistic model

which plots cumulative usage over time. Cumulative usage $\sum_i u_i$ should be monotonically non-decreasing, except from a rare event of reset (e.g. after an engine replacement). However, the recorded usage data was non-monotonic, with only the trend appearing to adhere to the expected behavior.

Two methods for estimating actual usage were considered: 1) development of a probabilistic model to reconcile the missing and inconsistent values, and 2) estimation of usage based on replacement of consumable parts. The two approaches are described in turn.

3.1. Probabilistic Model for Usage

MCMC is a sampling method that draws likelihood measurements from simulated random data conditioned by some predefined measured data. It utilizes a high dimensional search space where certain positions (representing a set of parameter values for the variables being estimated) have a greater likelihood than others. In order to search the parameter space, MCMC generates correlated random numbers from the current values by stepping to a new position of higher likelihood in parameter space by use of the gradient field (or probability density). This requires an initial guess to start the chain and is part of the conditioning.

3.1.1. First-Order Implementation

Figure 6 shows a simple probabilistic model used here: u_i and u_{i+1} are successive actual usages that are unknown, while M_i and M_{i+1} are the corresponding recorded measurements. The observations are shaded, following the convention of (Bishop, 2006)). The model is constructed as

$$M_i \sim \mathcal{N}(u_i, n_i), \tag{3}$$

so that actual usage u_i, distorted by noise n_i produces measurements M_i. The noise was measured in hours modeled as uniform between 0 and t_{max} (set to 50 to accommodate high errors encountered in the data)

$$n_i \sim \text{Uniform}(0, t_{max}). \tag{4}$$

The only assumption to model the unknown amount of usage between two measurements was that usage is monotonically non-decreasing. The usages were modeled as deterministic variables

$$u_{i+1} = u_i + b_i\Delta t_i, \tag{5}$$

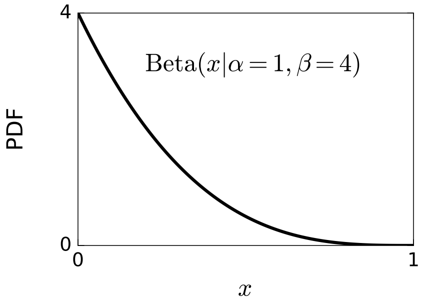

Figure 7. Usage factor modeled as a Beta(α,β) distribution with parameters $\alpha = 1, \beta = 4$.

where Δt_i was a known time interval between two samples, and b_i was a positive usage factor. It was important to keep these two factors separate rather than lumping them into a single parameter. As two parameters, they better related to the basic domain knowledge, allowing for time to be expressed in days, which improved the model's overall interpret-ability. In practice the usage was not recorded daily and the usage factor b_i, being a small fraction of a day, was modeled as the Beta distribution,

$$b \sim \text{Beta}(x|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}, \tag{6}$$

where $\Gamma(.)$ is the gamma function. The parameters α and β were empirically selected to 1 and 4, respectively, to model a larger likelihood to values closer to 0, as shown in Figure 7. Because of the recursive nature of Eq. 5, the first usage data point had to be modeled separately

$$u_1 = \mathcal{N}(M_1, M_1/2). \tag{7}$$

The separate treatment of the first usage point gave rise to undesirable behavior in cases where the first point contained unusually large error. The approach to fully address this problem is further explained in Section 3.1.2. However, because the probabilistic model aimed to maximize the likelihood of the entire fit it was less dependent on the initial estimate's accuracy as the number of samples increased. In addition, in cases where physically-impossible outliers were encountered (vehicle was used more than 24 hours per day), as evidenced by the condition

$$\frac{M_{i+1} - M_i}{\Delta t_i} > 1, \tag{8}$$

the noise was allowed to sample beyond the (0,50) limit (refer to the lines 30-32 in the listing provided in the Appendix).

The results of an estimation using the simple model implemented in Stan are shown in Figure 8). The orange line traces the means of the estimated distributions of usage $\hat{u}_i$.

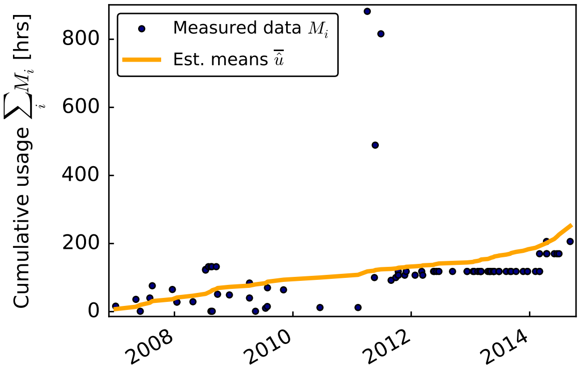

Figure 8. Usage estimation example

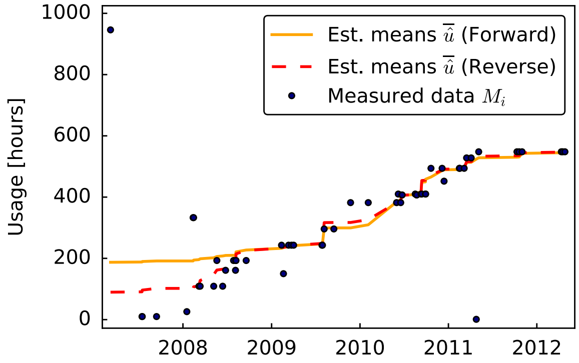

Figure 9. Forward and reversed model operated on the same data.

After running the program, each measured data point had a *probability density function* (PDF) assigned to it, built by the model. Every value in the PDF represented one of the accepted samples in the Markov chain. The mean of these PDFs was chosen to represent the estimate because it denotes the average value of all the samples. The mode was considered but not used due to the possibility of a value that may not have accurately represented the best estimate to that point being the value that the stepping method hit more often. This was one example of how the thinning and burn-in sampling parameters were useful. Thinning reduced samples where the stepping method got stuck, and burn-in removed earlier samples that may not have contributed to the final convergence.

While this simple form of a probabilistic network seems particularly well-suited for the problem at hand (featuring relatively few data points with potentially large errors), based on our literature search, we believe that it has not been used in the context of the usage estimation before. In addition, the assignment of the probability distributions, which is critical to model performance was described in detail. As other similar approaches ((Bishop, 2006) shows the connections between Kalman filters, HMMs, and other probabilistic networks), it returns a full probability distribution for the data, the posterior. This allows for a much fuller understanding of the uncertainty of the results. A script for the Stan implementation is provided in the Appendix.

3.1.2. Initial Value Dependence and Looping

The choice of a Beta distribution for the slope of the model described in the previous section imposed the limits for the cumulative usage, making it non-decreasing by design. One limitation of this approach was that the starting point has stronger impact than the subsequent measurements. It had no preceding data values to step up from so it had to be estimated some other way. It was estimated using a normal distribution with a user inputted standard deviation. This distribution could pick a value close to the data point or allow for variation when a bad initial value was recorded, however, the final fit was always biased by this initial estimate. When the first measured data point was greater than the successive points the overall fit tended to sit atop of the values near that data point. A good starting estimate led to a good overall fit. The quality of fit was determined by what was known. The only knowledge of this data was that it had to monotonically increase to make physical sense and that not all of the data was bad. If the estimates followed the monotonic rule and lied close to the plausible measured data, the fit was deemed valid. A reasonable data point was one that fell in line with the surrounding data in an increasing fashion. When the first data point was reasonable, the fit was reasonable. When the first data point was unreasonable, the fit was considerably less reasonable near that data point. Figure 9 shows two fits to some real usage data taken from the vehicle database. The forward fit worked from the beginning to end of the time data, and the reverse fit worked from end to beginning. Because the first data point was noisy, the reverse fit was a better estimate overall as it started from a reasonable data value.The orange line represents the average Bayesian fit evaluated from the first data point to the last. The red dashed line represents the average Bayesian fit evaluated from the last data point to the first. When evaluating in this direction, because the first point was noisy but the last point was not, the reverse fit looked more plausible.

It was hypothesized that the bias of the initial point would affect the final fit less when the model was run many times forward and reverse through the data and the results were averaged. The process was as follows: The first run estimated the first point with a normal approximation and fit the rest of the data. Then, taking the average value of the last data point's estimate, it started the process over again in reverse, this time starting with the endpoint. Once the reverse direction fit was computed, the program averaged that last point estimated (which was our starting point from the beginning) and continued again in the forward direction, this time having a different starting point than at first run. The only part of the code that needed to be changed in order to accommodate this condition was the deterministic expression shown in Eq. (5) which, in the reverse direction, was changed to

$$u_{i+1} = u_i - b_i \Delta t_i \tag{9}$$

and a feature to read the previous run's last estimate to start the next run. A new piece of data (boolean) which declared the direction of fit was added as well.

The fit behaved better where surrounding data existed than at the initial value and if the data were evaluated in the reverse direction (end to beginning), the estimate of the first value was more reasonable than the normal approximation described above. This was because the reverse direction would end on that first value and if it were an unreasonable value the preceding points would guide the fit better than if it started at that value. The idea was that this pattern would continue as more loops were made forward and backwards. That method did not show any significant decrease in bias, however, averaging the estimates of a single forward pass and a single (independent) reverse pass did decrease the bias experienced from the first point. It is important to note for this final method, the reverse fit was evaluated independently of the forward fit, unlike the looping procedure.

3.1.3. Testing with Synthetic Data

In order to further test the effectiveness of the overall fit, synthetic data was created so that true values could be known for a visual comparison. The objective of this estimator was to ultimately show the most plausible interpretation of noisy measurements with what prior knowledge of the data we had. Four cases were chosen to represent typical scenarios in the data and are shown in Figure 10. The first test case, Figure 10a, was intended to get a control set and see how the fit handled perfectly reasonable data. The crosses mark the true (synthetic) data. The dots mark the measured (observed) data. In this case the true data matched the measured data and so the final estimates showed much sharper peaks in their distribution. This verified that the model performed accurately with the most ideal case of measurements.

The second case, Figure 10b, included some noise added to the measurements. Notice here that some of the dots are not aligned with the crosses. The orange line, representing the mean trace of the posterior distributions, was almost identical to the first case. This showed that small instances of noise could be handled as if there was no noise.

The next case, Figure 10c, introduced larger noise to the true usage. Some measured points were above the true values and some were below, just like the second case. In this case the mean trace was still nearly identical to the first case, however, notice the uncertainty in the individual distributions has widened for those noisier observations.

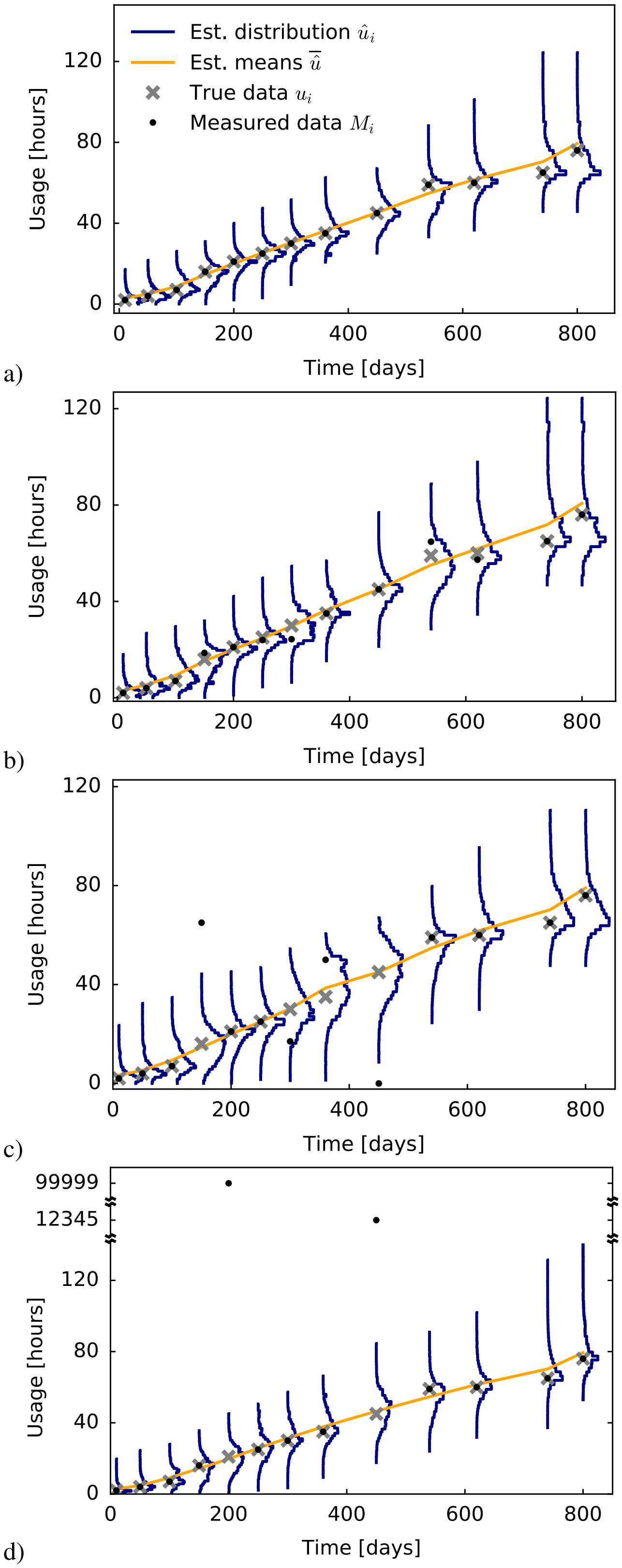

Figure 10. a) Case 1: Measured data equals true data. b) Case 2: Some noise added to measured data. c) Case 3: Large noise added to measured data. d) Case 4: Extreme outliers.

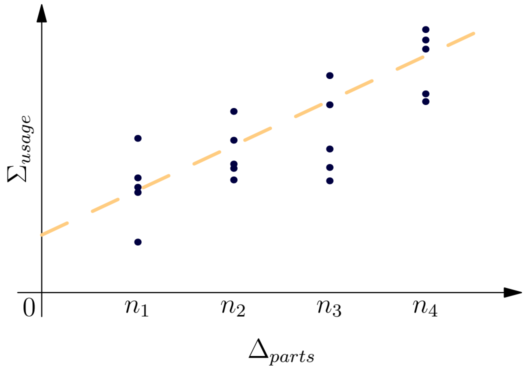

Figure 11. Hypothesized relationship for usage between maintenance events and number of replaced parts.

The fourth and final case, which introduced the effectiveness of the conditional free variable on noise when obvious outliers were detected, shows how the estimator handled two extreme outliers. It was not uncommon to see values like 12,345 and 99,999 in the data. Notice that in this final and extreme case the mean trace still did not move appreciably from the first case. The extreme outliers were basically ignored since such a large noise value was assigned to those estimates and the rest of the data fit just like the first case.

In practice all of these types of scenarios were prevalent in some combination or other in the measured data. Focusing on the mean trace between all these test cases shows the consistency this model had over different occurrences of noise. When analyzing these cases, this was the most important factor. The consistency of the fit among different noise cases indicated the consistency of the model. The model consisted of basic prior domain knowledge of how the data should behave, and an understanding of when values are too un-realistic to be modeled like the rest.

3.2. Replacements of Consumables

In order to supplement usage estimation, replacements of the consumable parts were analyzed. Any ground vehicle has a set of parts that degrade as a function of usage: tires, breaks, etc. In military tracked vehicles some examples of consumable parts include sprockets, road wheels, shock absorbers, track adjusters, support rollers, and track idler assemblies. It was expected for a vehicle to have replacement rates proportional to usage rates. Figure 11 shows a hypothesized relationship between number of replaced parts and usage increments associated with maintenance intervals.

Vehicle cost and maintenance histories were analyzed by sub-system (i.e a collection of components categorized by their collective purpose). Components with known use-based degradation, such as road wheels, were analyzed individually. Replacement frequency and failure rate differences among vehicles were used to test correlation between consumable component's maintenance and vehicle's usage.

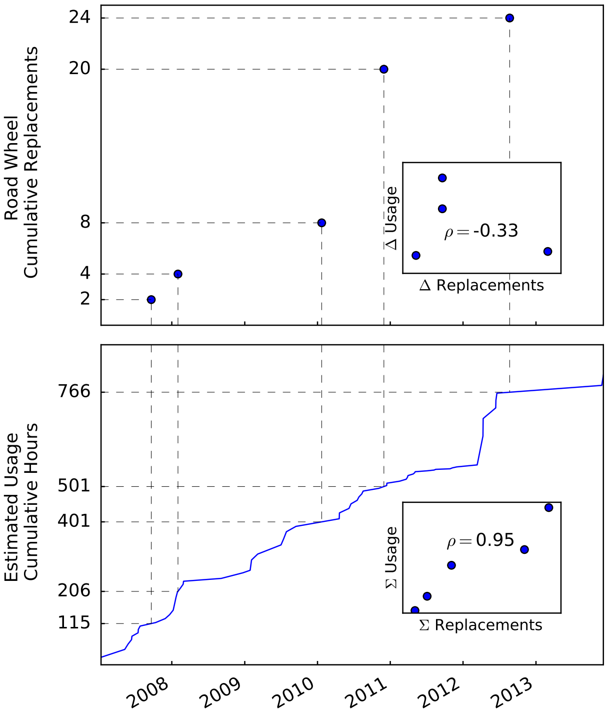

Figure 12. The two inset plots show the correlation between change in usage and change in replacements (top) and between cumulative usage and cumulative replacements (bottom).

Maintenance orders record the part replaced as well as the number of parts replaced. Figure 12 shows the cumulative replacement of road wheels over time and estimated cumulative usage, using the model described in Section 3. The inset plot on the top axis in Figure 12 shows the change in usage between maintenance over the change in number of replacements.

The data for the number of replaced parts was inconsistent over the fleet. The majority of data for these vehicles contained few maintenance records. The average usage between maintenance for the road wheel component example in Figure 12 was approximately 230 hours. The average usage range for the part of the fleet that was analyzed (1,247 vehicles) was approximately 223 hours.

Since the fleet average was less than the road wheel average it was expected that the vehicles in this analysis contained data with no maintenance events during the usage period. This is illustrated in Figure 13. The histogram shows the number of road wheel replacement across the fleet: 85% of vehicles did not have any road wheel replacement records in the maintenance data, ∼9% of the vehicles have 1-2 wheel replacement,

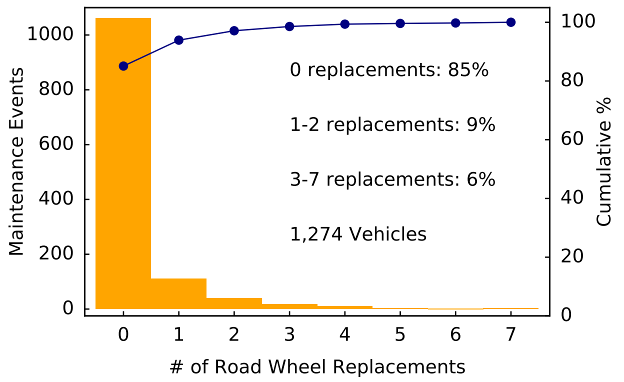

Figure 13. Only 17% of records for road wheels have one or more replacements associated with them.

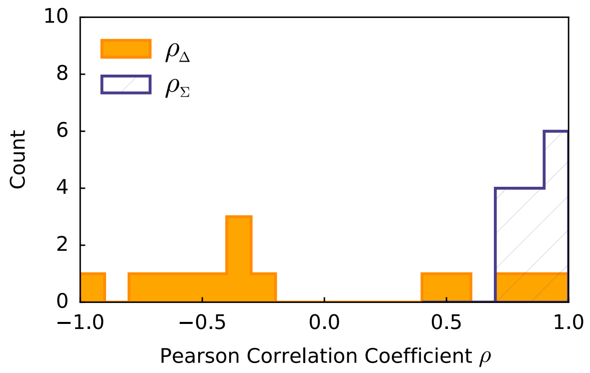

Figure 14. Pearson correlation coefficients for all vehicles that had usage records and replacements of road wheels.

etc. The maintenance history of most vehicles was insufficiently long to capture the usage patterns.

Of the several consumable components, road wheels had the most replacement data. It was expected, however, that there be few replacements over the usage history of the vehicle. The jagged edges of the usage estimation line in Figure 12 indicate where usage records existed because usage was estimated with a probabilistic model that followed a piece-wise linear approximation between all usage events. If there were replacements at each of those times the vehicle would have been under constant repair, indicating a larger underlying problem with the vehicle. This is why the change in usage to change in number of replacements did not correlate well. The inset plot on the bottom axis of Figure 12 shows cumulative usage over cumulative replacements. Here, the correlation coefficient was 0.95 indicating these values were highly correlated. Overall, usage was indicated by the number of replacements when long term accumulation of data was observed.

Figure 14 shows a count of the computed Pearson correlation coefficients for all vehicles that had usage records and replacements of road wheels. The average of the correlation between change in usage and change in replacements, ρ_Δ, was -0.07. The average of the correlation between cumulative usage and cumulative replacements, $\rho\Sigma$, was 0.87. Replacement data are more important to the maintainer than usage and so usage values were more likely to be entered incorrectly. Replacements occurred during maintenance events but usage readings were based on logging from the vehicle operators and so replacements were expected to be entered with greater care.

The vehicle shown in Figure 12 used the mean usage estimation from our probabilistic model in comparison to long term accumulation of replacements of the road wheel component. This was done for other vehicles as well. In Figure 12 matching usage and number of replacements in time is shown by dashed lines connecting the values between the axes. These pairs were how the inset plots were generated in that figure.

Figure 15 shows cumulative (Σ) usage versus cumulative number of replacements for fourteen vehicles. The slopes of these lines did not agree very well. In principle, one could use replacements of consumables to support usage estimation. There are several reasons we would see such variation in slope in this case. The usage estimation may not have been accurate due to bad data from the start for some vehicles. The replacement data may have been incorrect as well (seems likely for the bottom-right line that goes out to 80 replacements after around 100 hours of use). The operating conditions of different vehicles may vary greatly depending on their location too. Referring back to Figure 4, unless you know the circumstances of usage it is hard to relate it to replacements. Suppose these vehicles operated under different location-based conditions that were known to the user. The differences in slopes attribute to different conditions. If a set of vehicles were pre-classified by the relation between usage and replacements then replacement data for new vehicles, that are known to be part of a specific group, could aid in usage estimation. Using the slope for that class of vehicle would support likelihood estimates in the form of prior knowledge in the Bayesian usage estimator.

4. CONCLUSION

The focus of this study was to use improved methods in estimating long term accumulation of data that may be inconsistent or missing. Usage records were analyzed from military tracked vehicles and the approach to estimating usage based on noisy data was split into two parts. 1) development of a probabilistic model, and 2) estimation based on replacement of related consumable parts. The first method was a knowledge driven probabilistic model that estimated reasonable values for non-decreasing monotonic measurements that were inconsistent, incomplete, and sometimes missing in the data.

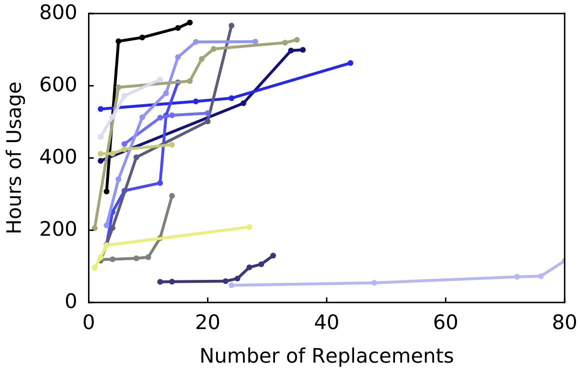

Figure 15. A comparison between hours of usage and number of road wheel replacements for fourteen vehicles.

The results were consistent for several cases of noise on the same data. We believe that this modeling approach, that requires minimal prior knowledge, has not been used for usage estimation. Because the ground truth was not available, we proposed and executed methods for evaluating performance of the model based on synthetic data in the absence of targets, including bi-directional passing through the data. The model was designed to estimate successive measurements and when applied to real usage data it was realized that noisy initial values biased the results. This was reconciled using an averaging method where the same data-set was sampled in a forward and reverse direction and the results averaged. This proved to significantly decrease the bias from a bad initial data point.

The second approach was intended to estimate usage as well but was found to be only supplementary at best. For the road wheel component, correlation between usage and replacements of consumables was very close to 0 from the maintenance to maintenance perspective, however, long term accumulation of data showed high correlation. This suggests that cumulative records of replacements of consumables related to usage can be used to aid in the estimation of usage itself.

ACKNOWLEDGMENTS

This work was made possible by the Department of the Navy, Office of Naval Research under Grant No. N00014-14-1-0789. The authors gratefully acknowledge the help of our colleagues and students from Rochester Institute of Technology: Timothy Murtaugh organized Excel files into a Microsoft SQL database, Scott Nichols help with formulating more intricate SQL queries, and Richard Price performed preliminary manual analysis of repairs. Their contributions are greatly appreciated.

DISCLAIMER

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Office of Naval Research.

REFERENCES

Andrieu, C., De Freitas, N., Doucet, A., & Jordan, M. I. (2003). An introduction to mcmc for machine learning. *Machine learning*, *50*(1-2), 5–43.

Barber, D. (2012). *Bayesian reasoning and machine learning*. Cambridge University Press.

Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., & Bengio, Y. (2010). Theano: a cpu and gpu math expression compiler. In *Proceedings of the python for scientific computing conference (scipy)* (Vol. 4, p. 3).

Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

Bussey, H. E., Nenadic, N. G., Ardis, P. A., & Thurston, M. G. (2014). Case study: Models for detecting low oil pressure anomalies on commercial vehicles. In *Annual conference of the prognostics and health management society 2014* (p. 13). PHM Society, http://www.phmsociety.org/node/1466/.

Cox, R. T. (1946). Probability, frequency and reasonable expectation. *American journal of physics*, *14*(1), 1–13.

Cox, R. T. (1961). *Algebra of probable inference*. John Hopkins University Press.

Das, S., Hall, R., Patel, A., McNamara, S., & Todd, J. (2012, June). An open architecture for enabling cbm/phm capabilities in ground vehicles. In *Prognostics and health management (phm), 2012 ieee conference on* (p. 1-8). doi: 10.1109/ICPHM.2012.6299529

Davidson-Pilon, C. (2015). *Bayesian methods for hackers: Probabilistic programming and bayesian inference*. Addison-Wesley Professional.

Denoeux, T. (2013). Maximum likelihood estimation from uncertain data in the belief function framework. *IEEE Transactions on knowledge and data engineering*, *25*(1), 119–130.

Diaconis, P. (2009). The markov chain monte carlo revolution. *Bulletin of the American Mathematical Society*, *46*(2), 179–205.

Ellerbrock, P. J., Shanthakumaran, P., & Halmos, Z. (1999). Development of new health and usage monitoring system tools using a nasa/army rotorcraft. In *Annual forum proceedings-american helicopter society* (Vol. 55, pp. 2337–2348).

Engel, S. J., Gilmartin, B. J., Bongort, K., & Hess, A. (2000). Prognostics, the real issues involved with predicting life remaining. In *Aerospace conference proceedings, 2000 ieee* (Vol. 6, pp. 457–469).

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis* (Vol. 2). Taylor & Francis.

Gelman, A., Lee, D., & Guo, J. (2015). Stan a probabilistic programming language for bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, 1076998615606113.

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*(6), 721–741.

Goebel, K., Daigle, M., Saxena, A., Sankararaman, S., Roychoudhury, I., & Celaya, J. (2017). *Prognostics: The science of making predictions.* CreateSpace Independent Publishing Platform.

Goodman, N., Mansinghka, V., Roy, D., Bonawitz, K., & Tarlow, D. (2012). Church: a language for generative models. *arXiv preprint arXiv:1206.3255*.

Gordon, A. C. (1991). *The development to production of an integrated health and usage monitoring system for helicopters.*

Gregory, P. (2005). *Bayesian logical data analysis for the physical sciences: A comparative approach with mathematica® support*. Cambridge University Press.

Heine, R., & Barker, D. (2007). Simplified terrain identification and component fatigue damage estimation model for use in a health and usage monitoring system. *Microelectronics Reliability*, *47*(12), 1882–1888.

Honaker, J., & King, G. (2010). What to do about missing values in time-series cross-section data. *American Journal of Political Science*, *54*(2), 561–581.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in science and engineering*, *9*(3), 90–95.

Jaynes, E. T. (2003). *Probability theory: the logic of science*. Cambridge university press.

Jones, E., Oliphant, T., Peterson, P., et al. (n.d.). Scipy: Open source scientific tools for python, 2001-2008b. *URL http://www. scipy. org*.

Jordan, M. I. (1998). *Learning in graphical models:[proceedings of the nato advanced study institute...: Ettore mairona center, erice, italy, september 27-october 7, 1996]* (Vol. 89). Springer Science & Business Media.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). Winbugs-a bayesian modelling framework: concepts, structure, and extensibility. *Statistics and computing*, *10*(4), 325–337.

McKinney, W. (2012). *Python for data analysis: Data wrangling with pandas, numpy, and ipython.* " O'Reilly Media, Inc.".

Metropolis, N., & Ulam, S. (1949). The monte carlo method. *Journal of the American statistical association*, *44*(247), 335–341.

Minka, T., Winn, J., Guiver, J., & Knowles, D. (2012). Infer .net 2.5. *Microsoft Research Cambridge*.

Moubray, J. (1997). *Reliability centered maintenance*. Industrial Press.

Nowlan, F., & Heap, H. (1978). *Reliability-centered maintenance* (No. AD/A066 579).

Oliphant, T. E. (2006). *A guide to numpy* (Vol. 1). Trelgol Publishing USA.

Patil, A., Huard, D., & Fonnesbeck, C. J. (2010). Pymc: Bayesian stochastic modelling in python. *Journal of statistical software*, *35*(4), 1.

Pearl, J. (1987). Evidential reasoning using stochastic simulation of causal models. *Artificial Intelligence*, *32*(2), 245–257.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

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

Plummer, M., et al. (2003). Jags: A program for analysis of bayesian graphical models using gibbs sampling. In *Proceedings of the 3rd international workshop on distributed statistical computing* (Vol. 124, p. 125).

Rabeno, E., & Bounds, M. (2009, March). Condition based maintenance of military ground vehicles. In *Aerospace conference, 2009 ieee* (p. 1-6). doi: 10.1109/AERO.2009.4839683

Shafer, G., et al. (1976). *A mathematical theory of evidence* (Vol. 1). Princeton university press Princeton.

Sikorska, J. Z., Hodkiewicz, M., & Ma, L. (2011). Prognostic modelling options for remaining useful life estimation by industry. *Mechanical Systems and Signal Processing*, *25*(5), 1803-1836.

Sivia, D. S. (1996). *Data analysis: a bayesian tutorial*. Oxford university press.

Team, S. D. (2016a). *Pystan: the python interface to stan, version 2.9.0. http://mc-stan.org.*

Team, S. D. (2016b). *Rstan: the r interface to stan, version 2.10.1. http://mc-stan.org.*

Theodoridis, S. (2015). *Machine learning: A bayesian and optimization perspective*. Academic Press.

Trammel, C., Vossler, G., & Feldmann, M. (1997). Uk ministry of defence generic health and usage monitoring system (genhums). *Aircraft Engineering and Aerospace Technology*, *69*(5), 414–422.

US-AMRDEC. (2016). *Ads-79e handbook for condition based maintenance system for us army aircraft.*

Vachtsevanos, G., Lewis, F., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent fault diagnosis and prognosis for engineering systems*. Wiley.

Zadeh, L. A. (1965). Fuzzy sets. *Information and control*, *8*(3), 338–353.

Zadeh, L. A. (1983). The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy sets and systems*, *11*(1), 197–198.

BIOGRAPHIES

Nenad G. Nenadic received his B.S. in Electrical Engineering from University of Novi Sad (Novi Sad, Serbia) in 1996 and his MS and Ph.D. in Electrical and Computer Engineering from University of Rochester (Rochester, NY, USA) in 1998 and 2001, respectively. He joined Kionix Inc. in 2001, where he worked on development of micro-electromechanical inertial sensors. Since 2005, he has been with Center for Integrated Manufacturing Studies (CIMS) at Rochester Institute of Technology, where he is currently a Research Associate Professor. His research interest include design, analysis, and monitoring of electromechanical devices and systems. He has two patents in electromechanical design and ten publications. He co-authored a textbook *Electromechanics and MEMS* and is a member of IEEE.

Christopher J. Valant received his B.S. in Physics from Rochester Institute of Technology (RIT), where he is currently working towards an M.S. in Big Data Analytics. He joined the Golisano Institute of Sustainability (GIS) in 2015. His research interests include applied machine learning, evolutionary algorithms, computer vision, simulations, optimization and programming.

Sean P. McConky received his B.S. in Industrial Engineering from Rochester Institute of Technology (Rochester, NY, USA) in 1999. He joined the Center for Integrated Manufacturing Studies (CIMS) at Rochester Institute of Technology in 2000, where he is currently a Senior Staff Engineer. His research interests include condition based maintenance, asset health monitoring, additive manufacturing, and remanufacturing. He holds one patent for dynamic display systems related to asset health monitoring. He was a member of the team awarded the Defense Manufacturing Excellence Award for work on lifecycle logistics support tools.

Michael G. Thurston received his B.S. and M.S. in Mechanical Engineering from Rochester Institute of Technology (Rochester, NY, USA) in 1988, and his Ph.D. in Mechanical and Aerospace Engineering from the University of Buffalo (Buffalo, NY, USA) in 1998. He is the Technical Director and Research Associated Professor at the Center of Integrated Manufacturing Studies at Rochester Institute of Technology. He formerly held positions in air conditioning system development at General Motor and Delphi, and as a Researcher at the Applied Research Laboratory at Penn State University. He holds 7 patents in the areas of air conditioning and asset health monitoring. His research interests include: sustainable design and production, condition based maintenance and prognostics, and asset health management. He is a member of the Society of Automotive Engineers, and was awarded the Boss Kettering Award for product innovation by Delphi.

APPENDIX

The goal of the appendix is to enable the reader to reproduce the simulations in the seleted computational environment. Rigourous comparison of the probabilistic frameworks was out of scope of the present project. In our experiments, which did not attempt to fully optimize frameworks, Stan marginally outperformed PyMC. The slight performance edge determined the selection. The language manual can be found at `http://mc-stan.org` along with an installation guide and section on getting started. Stan has several implementations available including RStan(Team, 2016b) for R users, and PyStan(Team, 2016a) for Python users. PySstan provides an interface to Stan, where some model code is written in $C++$ and implemented in Python (Figure 16). A $C++$ compiler is required to be available to Python during installation and at run-time. We used a Linux machine to run the probabilistic code because at the time it was most reliable.

```
usage_model = """
data {
int N; //number of observations
real M[N]; //observed usage of dimension N [hours]
real<lower=0> dt[N-1]; //delta time of dimension N
    -1 [delta days]
int F; //forward-reverse condition
}
parameters {
real<lower=0,upper=1> b[N-1]; //usage factor [
    usage/days]
real M0;
real<lower=0,upper=40> sigma0; //initial noise
real<lower=0,upper=40> sigma[N]; //noise
real n_outlier; //conditional noise parameter for
    large outliers
}
transformed parameters {
real u[N]; //estimated (real) usage

u[1] <- M[1] + M0;
for (i in 2:N) {
if (F==1) {
u[i] <- u[i-1] + b[i-1]*dt[i-1]; //forward case
}
else if (F==0) {
u[i] <- u[i-1] - b[i-1]*dt[i-1]; //reverse case
}
}
}
model {
M0 ~ normal(0,sigma0); //First data point
b ~ uniform(0,1);
for (i in 2:N) {
if (((M[i]-M[i-1])/dt[i-1])>1) {
M[i] ~ normal(u[i], n_outlier); // for clear
    outliers
}
else {
M[i] ~ normal(u[i], sigma[i]);
}
}
}
"""
# Compile model
m = pystan.StanModel(model_code=usage_model)
# Run Sampler
fit = m.sampling(data=data, iter=10000, chains=4)
```

Figure 16. Stan code example with PyStan implementation