

Copyright
by
Jonathan Robert LeSage
2013

The Dissertation Committee for Jonathan Robert LeSage
certifies that this is the approved version of the following dissertation:

**Energy storage-aware prediction/control for mobile
systems with unstructured loads**

Committee:

Raul G. Longoria, Supervisor

Maruthi R. Akella

Joseph J. Beaman

Dongmei Chen

Dragan Djurdjanovic

**Energy storage-aware prediction/control for mobile
systems with unstructured loads**

by

Jonathan Robert LeSage, B.S.M.E.; M.S.E.

DISSERTATION

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2013

Acknowledgments

While scholarly endeavors may externally appear as a solitary undertaking, in reality, the path of an academic remains flush with collaboration and assistance. My scholarly path has been no different, and I want to take the time to thank those who provided invaluable assistance. First and foremost, I would like to thank my advisor, Dr. Raul Longoria, for his steadfast guidance and counsel. Under his mentoring, I have grown tremendously in both my academic knowledge and confidence, and as a result, fostered an undying passion for research.

I also want to thank each of the members of my dissertation committee for serving as both research mentors and educators over the years. Considerable knowledge was garnered from courses and meetings with each committee member. Additionally, I would like to extend thanks to Bill Shutt for additional direction during my tenure at the Applied Research Laboratories.

Furthermore, I would like to thank the Office of Naval Research and the Electric Ship Research and Design Consortium for providing the funding for this project. Moreover, I want to thank Andrew Moore and Joey Groff from the Southwest Research Institute and Dr. Eric Krotkov from Griffin Technologies for their support with the Packbot discharge test field studies.

Countless friends have paved a path of work-life balance which has en-

hanced this process. In particular, I want to thank Donghwan Kim, Alexander Bleakie, and Albert Espinoza for providing both much needed feedback on ideas and aiding in the refinement of mathematical notation.

I also want to thank my parents and siblings for their love and continual support. In particular, I thank my parents for fostering my creativity and natural inquisitiveness which has ultimately helped me thrive. Finally, I want to thank my wife, Stephanie Kim LeSage, for her unwavering love and reassurance. Her perennial support has provided a beacon of serenity in the oftentimes turbulent world of academia.

Energy storage-aware prediction/control for mobile systems with unstructured loads

Jonathan Robert LeSage, Ph.D.
The University of Texas at Austin, 2013

Supervisor: Raul G. Longoria

Mobile systems, such as ground robots and electric vehicles, inherently operate in stochastic environments where load demands are largely unknown. Onboard energy storage, most commonly an electrochemical battery system, can significantly constrain operation. As such, mission planning and control of mobile systems can benefit from *a priori* knowledge about battery dynamics and constraints, especially the rate-capacity and recovery effects.

To help overcome overly conservative predictions common with most existing battery remaining run-time algorithms, a prediction scheme was proposed. For characterization of *a priori* unknown power loads, an unsupervised Gaussian mixture routine identifies/clusters the measured power loads, and a jump-Markov chain characterizes the load transients. With the jump-Markov load forecasts, a model-based particle filter scheme predicts battery remaining run-time. Monte Carlo simulation studies demonstrate the marked improvement of the proposed technique. It was found that the increase in computational complexity from using a particle filter was justified for power load transient jumps greater than 13.4% of total system power.

A multivariable reliability method was developed to assess the feasibility of a planned mission. The probability of mission completion is computed as the reliability integral of mission time exceeding the battery run-time. Because these random variables are inherently dependent, a bivariate characterization was necessary and a method is presented for online estimation of the process correlation via Bayesian updating. Finally, to abate transient shutdown of mobile systems, a model predictive control scheme is proposed that enforces battery terminal voltage constraints under stochastic loading conditions. A Monte Carlo simulation study of a small ground vehicle indicated significant improvement in both time and distance traveled as a result.

For evaluation of the proposed methodologies, a laboratory terrain environment was designed and constructed for repeated mobile system discharge studies. The test environment consists of three distinct terrains. For each discharge study, a small unmanned ground vehicle traversed the stochastic terrain environment until battery exhaustion. Results from field tests with a Packbot ground vehicle in generic desert terrain were also used. Evaluation of the proposed prediction algorithms using the experimental studies, via relative accuracy and $\alpha - \lambda$ prognostic metrics, indicated significant gains over existing methods.

Table of Contents

Acknowledgments	iv
Abstract	vi
List of Tables	xii
List of Figures	xiii
Chapter 1. Introduction	1
1.1 Active Challenges and Research Objectives	3
1.2 Overview of Methodology	5
1.2.1 Online Unsupervised Characterization of Loads	7
1.2.2 Model-based Prediction of Run Times	7
1.2.3 Online Mission Reliability Assessment	8
1.2.4 Real-time Energy Aware Control	9
1.3 Dissertation Overview	9
Chapter 2. Literature Review	11
2.1 Run-time Prediction Algorithms	11
2.1.1 Static Battery Run-time Maps	12
2.1.2 Data-Driven Prediction Methodologies	15
2.1.3 Model-based Run-time Prediction	16
2.1.3.1 Particle Filter-based Prognostics	20
2.1.3.2 General Markovian Stochastic Process	21
2.1.3.3 Particle Filter Algorithm	22
2.1.4 Metrics for Prognostic Fidelity Evaluation	24
2.2 Load Characterization Methodologies	28
2.2.1 Load Characterization and Forecasting	29
2.2.2 Forecast Monitoring	31

2.3	Dynamic Battery Models	32
2.3.1	Systems-level Battery Process	32
2.3.2	Modeling of Battery Physics	35
2.3.3	Lumped Parameter Battery Model	38
2.4	Probability of Mission Completion Assessment	41
2.5	Energy-aware Control Schemes	43
2.6	Summary	46
Chapter 3.	Online Stochastic Load Characterization	48
3.1	Jump Markov Process Modeling	51
3.1.1	Markov Chain Theory	51
3.1.2	Jump-Markov Process Realization	53
3.1.3	Characterization of a Markov Process	54
3.2	Gaussian Mixture Clustering	55
3.3	Unsupervised Model Selection	60
3.4	Integrated Algorithm	63
3.4.1	Ground Vehicle Load Characterization	64
3.4.2	Sample horizon cross-validation	67
Chapter 4.	Battery Remaining Run-time Prediction	70
4.1	Implementing Sequential Monte Carlo	71
4.1.1	Particle-based Evaluation of Remaining Run-time	75
4.1.2	Scalar Run-time Prediction	76
4.2	Battery Model for Prediction	77
4.3	Load Considerations for Algorithm Selection	81
4.3.1	Second-order Markov Load Simplification	81
4.3.2	Prediction Fidelity and Load Cluster Separation	83
4.4	Experimental Validation	86
4.4.1	Packbot Case Study	86
4.4.2	In-house Small Ground Vehicle Test Stand Study	94
4.4.2.1	DaNI Unmanned Ground Vehicle	94
4.4.2.2	Automated Terrain Environment for UGV Testing	96
4.4.2.3	Run-time Prediction Analysis	99

Chapter 5. Online Mission Reliability Assessment	106
5.1 Evaluation of Overall Mission Time	109
5.1.1 Task Time Estimation	110
5.1.2 Overall Travel Time Prediction	112
5.1.3 Characterization of Overall Mission Time	115
5.2 Mission Completion Assessment	115
5.2.1 Determining a Mission Process Statistical Model	116
5.2.2 Bivariate Representation of Mission Process	119
5.2.3 Online Bayesian Estimation of Correlation	120
5.2.4 Assessment of Mission Reliability	124
5.2.5 Online Restructuring of Mission Objectives	127
5.3 Experimental/Field Validation	128
5.3.1 In-house Small Ground Vehicle Test Stand Study	129
5.3.1.1 Experimental UGV Mission Process	129
5.3.1.2 Probability of Mission Completion Experimental Analysis	131
Chapter 6. Energy-Aware Control	138
6.1 Energy-aware model predictive control	139
6.1.1 Model Formulation and Nomenclature	141
6.1.2 Model Predictive Control with a Time-Varying Linear Perturbation Model	143
6.2 Applications to Unmanned Ground Vehicles	147
6.2.1 Longitudinal Ground Vehicle Model	148
6.2.2 Monte Carlo Simulation Study	151
Chapter 7. Conclusions	157
7.1 Summary and Directions for Future Work	158
7.1.1 Online Unsupervised Transient Load Characterization	158
7.1.2 Battery Remaining Run-time Prediction via the Particle Filter	159
7.1.3 Online Mission Reliability Assessment	161
7.1.4 Transient Shutdown Prevention Control	163
7.2 Significant Scientific Contributions	164

Appendices	167
Appendix A. Modeling, Estimation and Parameter Identification of Dynamic Battery Systems	168
A.1 Parameter Identification and Optimization	168
A.1.1 Battery Discharge Experiments	169
A.1.2 Parameter Optimization Routine	172
A.2 Identification of Shutdown Conditions	174
A.3 Online State-of-Charge Estimation	177
A.4 Nonlinear Observability of Battery Model	184
Appendix B. DaNI Unmanned Ground Vehicle Experimental Results	188
Appendix C. Univariate Gaussian Mixtures and Clustering	213
C.1 Statistical Mixture Model	213
C.2 Expectation-Maximization Algorithm	215
Bibliography	218
Vita	245

List of Tables

4.1	Packbot load clusters	90
6.1	Energy-aware control shutdown conditions	154
A.1	Discharge curve polynomial fit.	171
A.2	Optimized parameters for a 3000 mA-h NiMH battery pack.	174
B.1	DaNI UGV Discharge Experiments - Power Results	188
B.2	DaNI UGV Discharge Experiments - Locomotion Results	190
B.3	In-house UGV study terrain distribution percentages.	191

List of Figures

1.1	General source-load abstraction for prediction/control given limited source energy.	1
1.2	Summary of the proposed methodology for mobile system operation in stochastic environments.	6
2.1	Summary chart of battery prediction schemes in the literature.	12
2.2	Peukert remaining run-time prediction with experimental discharge data.	14
2.3	First-order model-based linear variance propagation.	18
2.4	Comparison of first-order propagation and unscented transformation.	19
2.5	Generic fault prognosis with the particle filter scheme	21
2.6	Single step illustration of particle filtering prediction.	23
2.7	Cumulative relative accuracy prognostic evaluation metric	26
2.8	$\alpha - \lambda$ prognostic evaluation metric	27
2.9	General schematic of the battery electrochemical reaction	33
2.10	Open circuit battery discharge curve and aging effects	35
2.11	Battery cell polarization effects which link terminal voltage with load current	36
2.12	Systems level battery models in the literature appropriate for run-time prediction.	36
2.13	Modified Thévenin equivalent circuit model of a generic electrochemical battery system.	39
2.14	Multivariate reliability analysis	42
3.1	Energy storage dependency on sensors/actuators.	49
3.2	Example robotic transient loads from field operations.	50
3.3	Jump-Markov model structure.	51
3.4	Illustration of steady state load Gaussianity.	56

3.5	Gaussian mixture model approximation to a non-Gaussian load distribution.	57
3.6	Clustering of measured load data into local Gaussian models. .	59
3.7	Unsupervised model identification via Akaike information criterion.	62
3.8	Gaussian mixture and jump-Markov transient load characterization methodology.	64
3.9	Example power data for GMJM fit.	65
3.10	Example GMM identification via AIC	66
3.11	Jump-Markov identification.	67
3.12	Example realization of GJM Process.	67
3.13	K-folds Cross Validation Methodology.	68
3.14	Sample horizon k-fold cross validation.	69
4.1	Particle filter forecasting of non-Gaussian states for battery RRT prediction.	71
4.2	Sequential importance resampling implementation of the particle filter for model-based battery RRT prediction.	73
4.3	Battery RRT distribution reliability analysis	76
4.4	Representative implementation of voltage bus regulation with battery system.	78
4.5	Battery load discharge test with robot vehicle drawing constant current loads on the voltage bus, resulting in a constant power load on the battery system.	79
4.6	Simple loading process for algorithm evaluation.	82
4.7	Basic load structure realizations with ranging cluster separation. .	83
4.8	GMJM/PF prediction realization for MC study.	84
4.9	Monte Carlo CRA prediction fidelity as dependent on $\Delta p^{(L)}$. .	85
4.10	Monte Carlo $\alpha - \lambda$ prediction fidelity as dependent on $\Delta p^{(L)}$. .	85
4.11	Packbot traversing terrain	87
4.12	Packbot normalized GPS data.	88
4.13	Packbot stochastic bus current loads.	89
4.14	Packbot cluster number identification.	90
4.15	GMJM characterization of Packbot loads at 15 minutes. . . .	91
4.16	Predicted Packbot battery voltage.	92

4.17	Statistical run-time prediction results for the Packbot.	92
4.18	Packbot prediction fidelity over prognostic horizon.	93
4.19	Laboratory unmanned ground vehicle.	95
4.20	Automated terrain environment for UGV run-time experiments.	97
4.21	DaNI UGV on laboratory terrains.	98
4.22	Convergence of UGV experiment statistics.	100
4.23	Experimental UGV load demands.	101
4.24	Experimental runtimes for twenty-five UGV discharge studies.	102
4.25	GMJM/PF prediction realization for the DaNI UGV at 20 minutes.	102
4.26	DaNI UGV prognostic horizon for experiment one.	103
4.27	Prediction fidelity over all UGV run-time tests.	104
4.28	Comparison of algorithm computation times for prediction.	104
5.1	Conceptualized mobile robotic system as a general stochastic process.	107
5.2	Integrated mission reliability assessment/decision algorithm.	108
5.3	Analytically computation of OTT via summation of normal random variables.	111
5.4	Analytically determining remaining drive time via cumulative velocity forecasting.	113
5.5	Monte Carlo simulation demonstration of normality of analytically predicted drive-time distribution.	114
5.6	Monte Carlo simulation of simplified stochastic throttle commands and terrain properties with deterministic battery charge.	116
5.7	Single mission realization from the Monte Carlo simulation study. (a) UGV Battery Voltage. (b) UGV Cumulative Distance.	117
5.8	Results of a Monte Carlo simulation study of a UGV operating in a stochastic environment.	118
5.9	Bivariate normal characterization of the Monte Carlo UGV study results.	120
5.10	Online correlation estimation via Bayesian inference	121
5.11	Jeffery's uninformative prior distribution.	122
5.12	Single Bayesian estimation of OBRT and OMT correlation via sequential updating using single set of simulation data.	124

5.13	Mean Bayesian estimation of OBRT and OMT correlation via sequential updating using MC simulation data.	125
5.14	Rotation/translations of mission random variable for mission probability computational efficiency.	125
5.15	Example <i>a priori</i> specified mission with three tasks and two potential mission plans: the desired mission and the contingency plan.	127
5.16	Online restructuring of mission plan to adopt a contingency plan via PoMC decision metric.	128
5.17	Cumulative longitudinal displacement of the DaNI UGV for all inhouse discharge studies.	130
5.18	(a) Measured vehicle velocity of the DaNI UGV via differential LIDAR. (b) Normalized histogram depicting the non-Gaussian velocity trends of the first three experiments.	131
5.19	Experimental results from UGV terrain discharge tests comparing discharge time and cumulative distance traveled with a bivariate normal distribution fit.	132
5.20	Ground truth PoMC for DaNI UGV missions.	133
5.21	Online predictions of PoMC for experiment number four given desired mission distances of 550m and 650m over the UGV run-time.	134
5.22	Probability of mission completion estimates for all discharge tests.	135
5.23	Relative accuracy of PoMC Estimates by desired mission. . . .	136
5.24	Relative accuracy of PoMC Estimates for all experiments. . .	137
6.1	Energy-aware model predictive control scheme.	139
6.2	Schematic depiction of the receding horizon control scheme for two steps in time.	141
6.3	Block diagram representation of the receding horizon control scheme.	147
6.4	Longitudinal vehicle model for source-aware MPC algorithm assessment.	149
6.5	Rolling resistance stochastic load structure.	150
6.6	Energy-aware MPC simulation study illustrating transient shutdown prevention.	151
6.7	Heuristic constrained-PID for comparison to energy-aware MPC.	153
6.8	Monte Carlo comparison study results of energy-aware control schemes.	154

6.9	Box plot comparison of energy-aware MPC gains.	155
6.10	Comparison of computational times required over all UGV simulations.	156
A.1	(a) Schematic of the current controlled discharge setup. (b) Laboratory realization of the discharge setup.	170
A.2	(a) Selection of discharge data illustrating the recovery effect returning to the open circuit voltage. (b) Polynomial fit of the open circuit voltage data.	171
A.3	Comparison of simulated optimal parameters and discharge voltage data of a NiMH battery pack.	173
A.4	Battery Protective Circuitry which enforces voltage shutoff and over-current protection.	175
A.5	(a) Ramped voltage decay of “battery” system. (b) Transient drop of the DC-DC boost converter output.	176
A.6	Methodology for evaluation of shutdown characteristics of a battery power converter/protective circuit.	176
A.7	Experimentally measured shutdown voltage probability density function for the XP JCL30 protective circuit.	177
A.8	Overview of extended Kalman filter estimation scheme.	181
A.9	Discharge test data for estimator validation.	184
A.10	State of charge estimation via the extended Kalman Filter and estimator error.	185
A.11	State of charge variance used for initialization of the particle filter algorithm.	185
B.1	DaNI UGV Experimental Measurements	189

Chapter 1

Introduction

With the advent of reliable and robust rechargeable energy storage systems, e.g., secondary cell batteries, ultra capacitors, flywheels and fuel cells, energy storage system usage has proliferated over the past decade [1]. For instance, the global market for Li-ion cells is expected to jump nearly 350% in a span of just four years, with the worldwide demand rising from 5.4 GWh in 2011 to 24.2 GWh in 2015 [2]. Mobile electronic systems, ranging from small-scale consumer electronics and mobile robotics to behemoth electric warships account for significant/varied demands on energy storage technologies. However, energy storage devices inherently hold limited charge, and as such, isolated operation of such mobile electronic systems remains bounded in time.

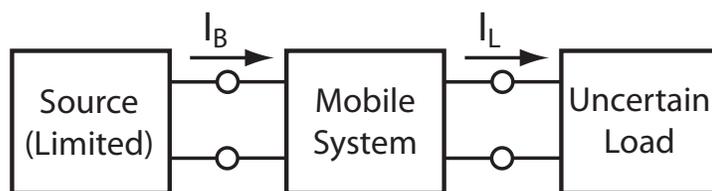


Figure 1.1: General source-load abstraction for prediction/control given limited source energy.

In many emerging applications, such as mobile robotics for reconnaissance in rugged environments and electric vehicles for commuting in dense

traffic, the power demands, imposed on the energy storage systems, display significant transient jumps and marked stochasticity due to operation in uncertain environments. For example, an unmanned ground vehicle moving from a non-deformable surface to a deformable one, such as snow, can result in an instantaneous jump in required power [3]. Transient jumps of 135W (a ratio of 58.7% of total power capacity) measured for the Polar navigator robot, illustrate the potential significant transients imposed on an energy-storage system [4]. Large mobile systems, such as next-generation electric ships with total generation capabilities in excess of 78 MW, can accommodate hybrid energy storage systems composed of synergistic ultracapacitors and battery banks [5]. Resultantly, significant power transients, such as the 25 MW required for a free electron laser pulse [6], draw from power-dense ultracapacitors as apparent energy in electrochemical batteries is power rate-limited [7]. However, in smaller scale applications, such as electric vehicles and particularly with ground robotics, the battery remains the de facto standard power source due to high energy densities, cost, and simplicity [8]. As a result, the onboard battery systems, in these applications, is directly subject to power transients and load stochasticity.

In response to the continued increase in battery-based mobile system deployment, research into energy source-aware prediction and control has continued to rise. The number of publications/patents with keywords of “battery,” “mobile system”, and “prediction”/“control” have increased such that publications over the decade of 2001 - 2011 (approximately 16,700) are outnumbered

by the number of cumulative publications since 2012 (approximately 17,200) according to a Google Scholar search. While the subset of the reviewed literature presents a variety of contrasting prediction/control methods, the unifying theme is the increased systems perspective where the battery and system are considered jointly [9, 10]. Resultantly, the reported battery/system prediction and control methods have increasingly adopted systems-level theory, such as condition-based maintenance for battery run time prognosis [11] and model-based supervisory control for battery energy optimization [10].

1.1 Active Challenges and Research Objectives

Considerable challenges remain for battery run time prediction/control for mobile systems operating in stochastic environments. For small-scale mobile vehicles, particularly field robotics where the battery is the dominant source of energy, significant load transients result in unexpected vehicle shutdown [12]. Existing methods presented in the literature have only recently begun to address the need for transient shutdown consideration in prediction and control [13]. As a result, a number of active challenges for mobile system run time prediction/control, which have either been disregarded or proposed for future work in the literature, remain:

- **Application-dependency of prediction algorithms:** Numerous battery remaining run time algorithms have been proposed over the past century, ranging from analytical Peukert's law [14] to model-based particle filter prediction [11]. While these algorithms range significantly in

computational complexity [15], no clear study has indicated the trade-off between power load characteristics and prediction accuracy.

- **Prediction of battery transient shutdown:** While mobile systems inherently operate with stochastic loads [16], existing battery remaining run time prediction schemes ignore load transients or ignore battery terminal voltage shutdown [17]. Consequentially, these algorithms disregard transient shutdown conditions, where transient load jump triggers battery protective circuitry shutdown [18].
- **Transient load history characterization:** For prediction of transient battery shutdown, the transient characteristics of the power load history must be modeled. Existing battery run time algorithms that consider transients require *a priori* characterization of loads [19]. However, offline experimental characterization of all potential power loads, for a mobile system in an unstructured environment, remains infeasible for all but the most simple applications [20].
- **Battery-based mission planning/assessment:** To consider the constraint of available onboard energy, mission and path planning algorithms have been developed which consider constrained energy [21]. However, for battery systems, the apparent available energy reduces for high current loads due to the rate-capacity effect. Resultantly, mission energy and available energy remain dependent processes which invalidate direct energy consumption comparison methods.

- **Transient shutdown prevention:** To optimize battery usage, supervisory control schemes have been developed which consider battery and drivetrain dynamics [22]. While these methods preserve battery state-of-charge, considerations of transient shutdown remain neglected.
- **Experimental analysis of stochastic energy-constrained mobile systems:** The battery discharge process of a mobile system operating in an unstructured environment is inherently stochastic. However, in the existing literature, validation of stochastic prediction methodologies for mobile systems have only considered single discharge experiments [13, 15, 17]. However, demonstration of statistically significant algorithm improvement requires numerous repeated studies [23].

The objectives of this dissertation are to expand the scope of existing battery prediction/control methods and to address the stated challenges. As discussed, a particular focus is paid to small ground vehicles due to their susceptibility to significant transient load variation. The following section briefly introduces the proposed methodology which addresses the discussed challenges.

1.2 Overview of Methodology

While the past decade has seen considerable growth in battery power mobile systems, energy-aware prediction/control in the literature have only just begun to address the need for transient power load considerations. As a

result, the goals of this work are: to broaden battery remaining run time prediction algorithms to include stochastic/transient load characteristics, reevaluate the relationship between demanded mission energy and available onboard energy, and investigate control methodologies that prevent battery transient shutdown conditions. While the discussion in this dissertation is applicable to any mobile system with constrained energy, a selective focus is given to small scale applications, particularly unmanned ground vehicles (UGV). As discussed, UGVs experience significant load transients due to varied terramechanical interactions, often while operating with only a single battery source.

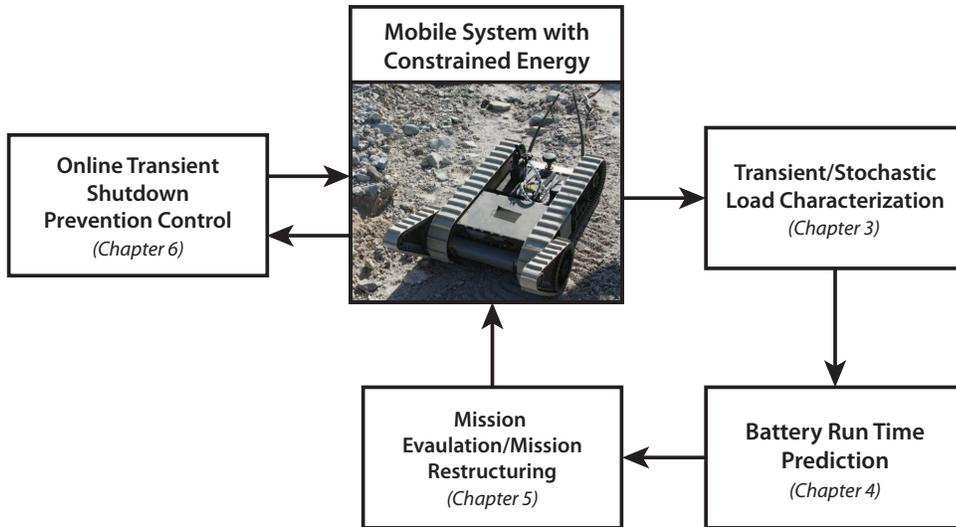


Figure 1.2: Summary of the proposed methodology for mobile system operation in stochastic environments.

To address the proposed goals of this work, several methods are proposed over the course of this dissertation, which are summarized in Figure 1.2. As such, each method is briefly introduced, and the active challenges that each

contribution addresses are elucidated.

1.2.1 Online Unsupervised Characterization of Loads

Mobile systems with finite onboard storage commonly operate in unstructured environments where load transients/statistics remain *a priori* unknown [12, 16]. To identify transient and stochastic load characteristics during mobile system operation, an online self-supervised load clustering algorithm is proposed in Chapter 3. The proposed methodology clusters measured power loads using a Gaussian mixture modeling scheme, where the number of load clusters is determined via model selection based on the Akaike information criterion. Upon identification of the transient loading clusters, maximum likelihood estimation is used to fit a jump-Markov chain. As a result, the integrated Gaussian mixture and jump-Markov model provides a stochastic characterization of power loads which can be used for prediction and load forecasting.

1.2.2 Model-based Prediction of Run Times

The battery discharge process which occurs during mobile system operation can be viewed as a fault with failure occurring upon system shutdown. As such, the particle filter, which has found widespread adoption in engineering fault prognosis [24], was adopted for battery remaining run time prediction. Provided a model of the battery dynamics, the particle filter allows for nonlinear/non-Gaussian state prediction and accounts for transient battery voltage shutdown conditions. Resultantly, a modified sequential importance

resampling particle filter routine is proposed in Chapter 4 that predicts the battery remaining run time probability density function. Finally, transient loads, as characterized by the jump-Markov chain, are used to forecast future expected power loads.

1.2.3 Online Mission Reliability Assessment

Mobile systems typically have objectives to achieve during operation (or during a mission), such as a completing a morning commute in an electric vehicle or a surveillance mission for a ground robot. These mobile systems, however, remain constrained by finite onboard energy storage, and as such, a finite probability exist for required mission energy exceeding available energy. Furthermore, in the case of battery systems, the available energy and mission energy remain correlated through the rate of power consumption. As a result, a bivariate characterization is proposed in Chapter 5 for evaluation of the probability of mission completion. An online Bayesian correlation estimation scheme is proposed to determine the bivariate relationship between mission time and battery run time. Integration over the region of failure (where mission time exceeds battery run time) yields an estimate for the probability of mission failure. Finally, a mission restructuring algorithm is proposed. When the primary mission probability of completion drops below a prespecified threshold, a contingency plan (such as aborting the mission) can be adopted.

1.2.4 Real-time Energy Aware Control

Mobile systems operating with only batteries remain subject to transient shutdown conditions [18]. Significant transient load demands, such as those encountered during ground robot locomotion in loose soil [25], can result in considerable terminal voltage reduction which triggers protective circuitry shutdown [26]. To preclude these transient shutdown scenarios, a constrained model predictive control methodology is proposed in Chapter 6 that integrates both the battery and drivetrain dynamics of a ground robot. Using transient load forecasts, the predictive control scheme adjusts the desired supervisory/teleoperated control command to ensure satisfaction of the battery terminal voltage constraint.

1.3 Dissertation Overview

The remainder of this work is arranged as follows. In Chapter 2, the literature concerning mobile system operation time prediction/control is reviewed with a particular emphasis on battery-based systems. In Chapter 3, the transient load characterization scheme is proposed and demonstrated with experimental data. In Chapter 4, a model-based particle filter scheme is introduced for online battery remaining run-time prediction. Results from two experimental studies involving ground robots are also presented which demonstrated predictor efficacy.

In Chapter 5, a method evaluating the probability of completing a mission/driving a distance is proposed which integrates methods from Chapters

3 and 4. This probability of mission completion algorithm is then experimentally demonstrated with a ground robot study. In Chapter 6, a transient shutdown prevention control scheme is introduced. Finally, in Chapter 7, the contributions of this work are summarized and directions for future research are discussed.

Chapter 2

Literature Review

Energy storage aware prediction/control and mission feasibility assessment remain multifaceted problems which have been explored in the literature with varying degrees of complexity. While some researchers have integrated some aspects of load characterization and prediction, these studies persist as exceptions rather than the norm. As such, the prior art is subdivided and discussed independently with cross disciplinary studies noted. Firstly, existing methodologies for energy storage prediction are explored. As a result of the inherent nonlinear and dynamic complexities, the battery system serves as the paradigm on which prediction algorithms are based, and as such, battery prediction is explored in detail. A review of statistical characterization techniques is presented for the modeling of stochastic power loads. Finally, literature detailing battery-aware predictive control schemes is discussed.

2.1 Run-time Prediction Algorithms

Due to the continually improving energy density of battery systems (with current military grade Li-ion cells reaching densities of 250 W-h/kg [27]), battery systems remain one of the prime candidates for mobile energy

storage [8]. Furthermore, due to the dynamic complexities of electrochemical systems and the ubiquitous adoption of battery-based energy storage, the mobile energy storage remaining run-time (RRT) prediction literature focuses on battery systems. For mobile systems operating with solely battery energy, prediction of the RRT provides a metric to aid in ensuring the safe return of a vehicle. A detailed review of battery RRT literature was conducted, and existing battery RRT algorithms were found to be divisible into three overarching groups: static RRT maps, model-based predictors, and data driven predictors.

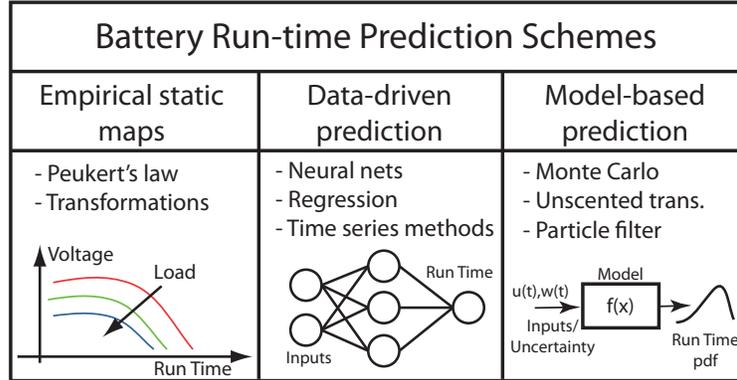


Figure 2.1: Summary chart of battery prediction schemes in the literature.

2.1.1 Static Battery Run-time Maps

Static RRT mapping algorithms rely on a longstanding empirical relationship known as Peukert's law [28], which was originally formulated experimentally in 1897 [14]. Fundamentally, Peukert's law is a current counting technique with a polynomial modifier on the battery load current. The polynomial current modifier empirically models the rate-capacity effect which result

from the internal diffusion process of ions in a battery cell [29]. Contemporary modifications [30–32] of the constant current load relationship expand Peukert’s law to include piecewise constant loading,

$$L = \frac{C}{\sum_{k=1}^m \alpha_k I_k^b} \quad (2.1)$$

where I_k are constant current loads belonging to some finite set $I_k \in I_1, I_2, \dots, I_m$ and α_k are the duty cycles of the respective current loads. Furthermore, L is the remaining life in hours, C is the battery capacity in A-h, and $b > 1$ is some experimentally determined finite scaling coefficient. As equation (2.1) is analytical, uncertainty analysis provides an estimate of the Peukert prediction variance [19]. Peukert prediction uncertainty is given as follows,

$$\sigma_L^2 = \bar{I}^{-2b} \sigma_C^2 + b^2 \bar{C}^2 \bar{I}^{-2(b+1)} \sigma_I^2 \quad (2.2)$$

where \bar{C} and σ_C^2 are the current mean and variance of the battery capacity in A-h, respectively. Furthermore, \bar{I} and σ_I^2 are the mean and variance of the battery current draw.

Other modern analytical methods [33–35] utilize algebraic transformations to exploit a region of linearity in the discharge process. These transformations remove the denominator polynomial term of Peukert’s law, and subdivide the time discharge curve into an exponential region and a linear region. Linear regression of the linear region of the transformed function provide for RRT prediction. To further extend the prediction capabilities of Peukert’s

law, researchers further modify equation (2.1) to accommodate additional inputs. These added empirical relationships allow for additional prediction inputs, such as temperature [36], current duty cycles [30–32,37], battery age [38], and current SOC [36,38]. In low rate discharge systems ($> 1A$) with purely constant discharge profiles ($m = 1$), Peukert’s law has been shown to accurately predict RRT of a battery system with lead-acid chemistry [28].

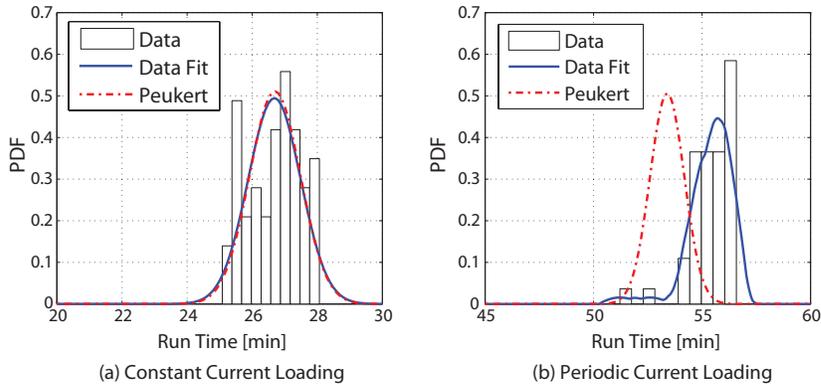


Figure 2.2: Comparison of Peukert’s run-time predictions made at 50% run-time using discharge data from [39]. (a) Constant current [4 A] discharge time. (b) Periodic current discharge [0-4 A]

While the static mapping techniques discussed above have been experimentally demonstrated under certain conditions, e.g. with lead-acid chemistries under constant current/temperature discharge, the methods fail more generally. Experimental studies [28] found Peukert’s law does not accurately predict RRT for Li-ion or NiMH batteries, and even fails with lead-acid chemistries when subject to stochastic loads. For mobile systems operating in field environments, battery power loads remain inherently stochastic due to both driver and environmental uncertainties [4, 12, 16]. Furthermore, modern mobile systems,

such as electric vehicles and unmanned ground robots, rely on high specific energy chemistries such as NiMH or Li-ion to lengthen operation time between charges [40]. Implementation of Peukert’s law with Li-ion constant/periodic discharge data [39], illustrated in Figure 2.2, further substantiates the results of [28]. Resultantly, RRT via empirical methodologies, such as Peukert’s law, demonstrates erroneous predictions for mobile system which experience transient loading.

2.1.2 Data-Driven Prediction Methodologies

To overcome limitations associated with static mapping methods, data-driven, or soft computing, approaches have been proposed to predict battery RRT using purely measurements. In the literature, data-driven RRT prediction for batteries falls into two distinct categories: battery state-of-charge (SOC) forecasting and discharge process learning via artificial intelligence techniques. Battery SOC, which is discussed in detail in Section A.3, indicates the remaining charge in a particular cell.

Correspondingly, provided online estimates of SOC, linear regression can forecast SOC to determine the time of cell exhaustion ($SOC = 0$) [41]. A least squares regression relating SOC to future run-time yields $SOC(t) = \phi_1 t + \phi_0$. Provided mobile system shutdown occurs at zero SOC, the predicted RRT can be expressed analytically,

$$L = -\frac{\phi_0}{\phi_1}. \tag{2.3}$$

While ample techniques exist for data-driven forecasting, which are discussed

further in Section 2.2, only nonlinear regression [41] and ARMA forecasting [42] have been presented in the literature for SOC RRT forecasts. As a physics-based battery model remains necessary for online SOC estimation, purely data-driven forecasting techniques discount the *a priori* known physics and suffer under transient loading scenarios [11].

Contrastingly, artificial intelligence methodologies have been employed to ascertain a nonlinear relationship between measured battery inputs and RRT [43]. In particular, an artificial neural network (ANN) was trained with 87 high fidelity battery discharge data sets. These discharge tests included measurements of battery current loads, terminal voltage and remaining discharge time [43]. Once trained, the ANN was shown to predict mean RRT with errors on the order of $\pm 10.55\%$, under the assumption that future loads are known [43]. For mobile systems of interest to this work, the future load history cannot be known with certainty [12]. Furthermore, for a 2.5 hour mission, errors with the ANN method would result in RRT uncertainty of ± 15.8 minutes. As a result, existing data-driven battery run-time techniques cannot be used in the field and do not adequately capture transient effects [41].

2.1.3 Model-based Run-time Prediction

As discussed, the static mapping methods and purely data-driven schemes fail to accurately predict battery RRT with stochastic loads due ignorance of transient/dynamic battery effects. The battery electrochemical system, which is introduced in detail in Section 2.3.3, exhibits transient load dependent ef-

fects, such as the rate-capacity effect and recovery effects [44]. Resultantly, prediction including transients requires a model basis which captures the dynamic effects of the battery system [24]. Furthermore, systems-level representations of the battery discharge process remain nonlinear, which is further elucidated in Section 2.3.3. As such, model-based prediction schemes must accommodate nonlinear dynamic models. To ensure computational tractability for online algorithm deployment, the discussion is restricted to discrete time stochastic systems [45]. Generally, a stochastic nonlinear discrete system is expressed as,

$$x_{k+1} = f(x_k, u_k, \omega_k) \quad (2.4)$$

where x_k and u_k are the system state vector and the system input at time, k , respectively. Furthermore, ω_k is the zero mean Gaussian process noise, and $f(\cdot)$ is the nonlinear discrete system which describes the state evolution. A stochastic continuous time nonlinear dynamic system can be discretized to the standard form of equation (2.4) via Euler-Maruyama discretization [46, 47].

Finally, as stochastic loading encountered in unstructured environments cannot be guaranteed to be Gaussian, the prediction scheme should account for non-Gaussian loading conditions. Provided a mathematical representation of the battery discharge process, several techniques exist for prediction of future states and RRT: *a priori* extended Kalman prediction [48, 49], unscented transformations [50, 51], batch Monte Carlo [19, 52], and sequential Monte Carlo (particle filtering) [15, 53–55].

Assuming only Gaussian state uncertainty and process noise, linearization of a nonlinear dynamic model, in equation (2.4), provides for prediction of future states/uncertainty. Furthermore, if the model uncertainty is restricted to be Gaussian, only the first two statistical moments, the state means and covariance, must be predicted [49]. This linear prediction, known as first order propagation, forms the basis of the prediction step of the extended Kalman filter [48]. Schematically, the first order propagation methodology is illustrated in Figure 2.3.

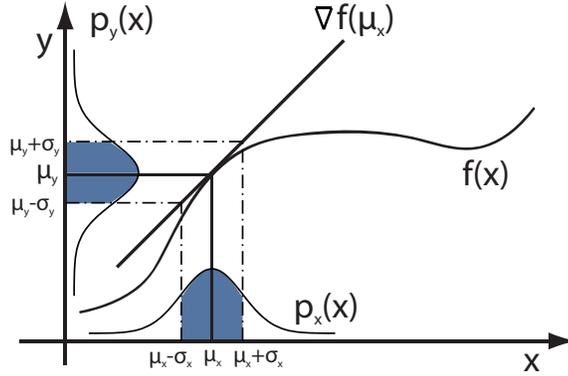


Figure 2.3: Illustration of model-based linear variance propagation via first-order Taylor expansion. Adapted from [56].

To predict the one-step ahead state and variance, the nonlinear transformation of equation (2.4) is linearized via a first-order Taylor series expansion [57]. Linearization of the nonlinear state function produces the linear state and noise matrices, given by $A_k = \nabla_x f(x_k, u_k, \omega_k)$ and $E_k = \nabla_\omega f(x_k, u_k, \omega_k)$, respectively. The first-order mean state prediction is given by equation (2.4), where x_{k+1} is the one step ahead mean state forecast. Correspondingly, the

state covariance prediction is given by,

$$P_{k+1} = A_k P_k A_k^T + E_k Q_k E_k^T \quad (2.5)$$

where P_{k+1} is the predicted covariance matrix and Q_k is the variance of the process noise.

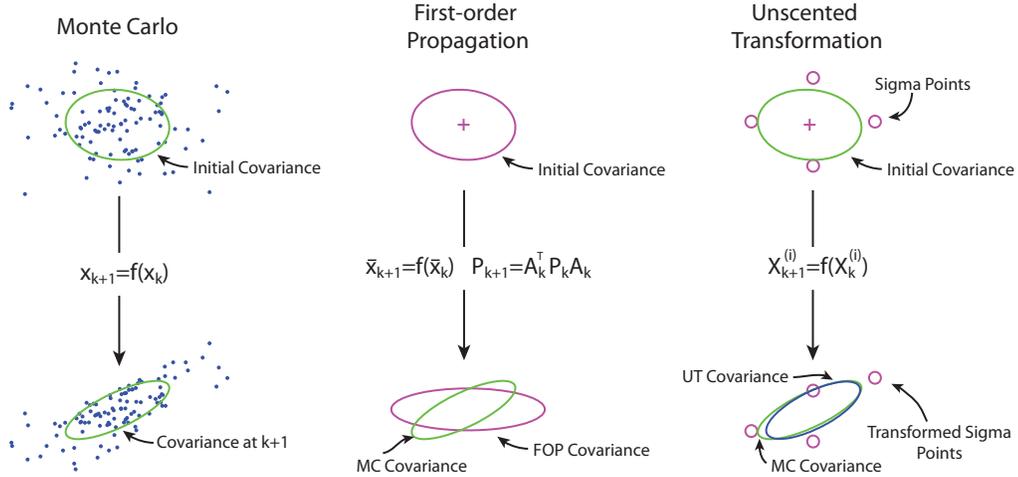


Figure 2.4: Comparison of model-based prediction methods. Adapted from [50].

Contrastingly, the unscented transformation (UT), which forms the basis for the prediction step of the unscented Kalman filter (UKF), replaces the first-order propagation prediction of the EKF. The UT, as compared to other model-based prediction techniques, is demonstrated in Figure 2.4. Rather than perform local linearization, a small set of significant particles, or sigma points, propagate through the nonlinear model of equation (2.4). The set of sigma points is composed of $2i + 1$ values, where i is the number of model inputs and model states [58]. These sigma points approximate a third order

Taylor series propagation of Gaussian state uncertainty through a nonlinear model [50]. Resultantly, the UT outperforms first-order propagation particularly when subjected to stark nonlinearities [59, 60]. Details of sigma point generation are provided in [61].

While the presented first-order propagation and unscented transformation methods require little in terms of computational complexity, the lack of non-Gaussian state prediction hinders prediction of complex systems [24]. Furthermore, predicted distributions of interest in this work, such as battery RRT in stochastic environments, have been shown to exhibit non-Gaussianity [11]. As such, the remaining discussion of model-based prediction methods is restricted to nonlinear/non-Gaussian methods.

2.1.3.1 Particle Filter-based Prognostics

The particle filter has seen increased utilization in recent years, beyond the initial proposed scope of online state estimation. In particular, the PF methodology has been altered for long-term model-based predictions, or prognostics, particularly in the field of condition-based maintenance [24, 55, 62]. Consequentially, as the PF allows for nonlinear dynamic models and provides for non-Gaussian predicted state representation, the PF remains the standard for model-based fault prognostics [63]. In Figure 2.5, the particle filter prediction scheme for system prognostics of a generic fault is presented.

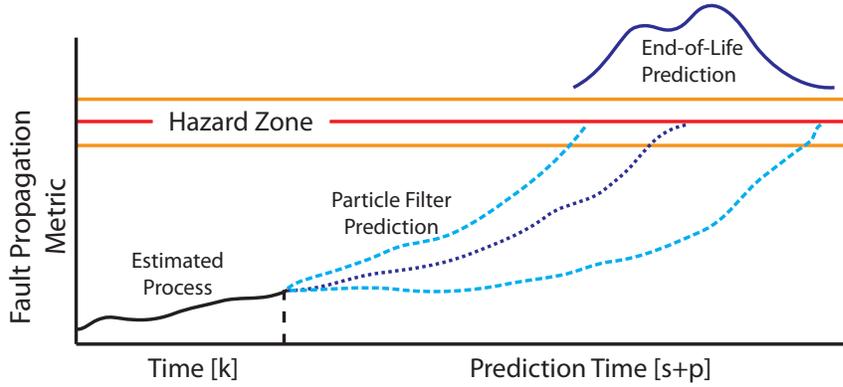


Figure 2.5: Generic fault prognosis with the particle filter scheme. Adapted from [24]

2.1.3.2 General Markovian Stochastic Process

Rather than modeling individual realizations of a stochastic process to predict remaining run-time, a Markov process representation models the evolution of the state probability density functions (PDF) [64]. Resultantly, the propagation of a PDF through a general nonlinear system can be expressed as, $P(x_k|x_{0:k-1})$, which is the conditional probability of the current state given the joint probability of all prior states, $x_{0:k-1}$. The probability of the joint conditional PDF, $P(x_k|x_{0:k-1})$ collapses to $P(x_k|x_{k-1})$ as a result of the celebrated Markov assumption [24]. For prediction given the Markov process representation, the joint probability of all the predicted states $P(x_{k:k+n})$ is given by the product of the conditional transition probabilities, as shown,

$$P(x_{k:k+n}, z_{1:k}) = P(x_k|z_{1:k}) \prod_{i=1}^n P(x_{k+i}|x_{k+i-1}) \quad (2.6)$$

where n is the prediction interval and $P(x_k|z_{1:k})$ is the current state estimate given the $z_{1:k}$ process measurements. To ascertain the marginal probability

density function of the final state prediction, $P(x_{k+n}|z_{1:k})$, the intermediate states must be marginalized [64]. Marginalization of the prediction states gives the p-step ahead PDF prediction, as follows,

$$P(x_{k+n}, z_{1:k}) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} P(x_k|z_{1:k}) \prod_{i=1}^n P(x_{k+i}|x_{k+i-1}) \prod_{i=0}^n dx_{k+i} \quad (2.7)$$

Evaluation of the marginalization integrals is analytically intractable and computationally complex to evaluate via numerical batch processing [64]. As a result, sequential prediction is adopted where the predicted state PDF, $P(x_{k+1}, z_{1:p})$, becomes the prior PDF, $P(x_k|z_{1:p})$, for the subsequent prediction. Mathematically, sequential prediction is expressed via the Chapman-Kolmogorov equation,

$$P(x_{k+1}, z_{1:p}) = \int_{-\infty}^{\infty} P(x_k|z_{1:p})P(x_{k+1}|x_k)dx_k, \quad (2.8)$$

where p is the initialization time of the prediction. Under certain assumptions, equation (2.8) becomes analytically tractable. For example, when assuming a linear system process, normally distributed states and Gaussian process noise, equation (2.8) reduces to the Kalman prediction equation [48]. However, for general non-Gaussian stochastic loading and non-normally distributed states, the Chapman-Kolmogorov equation remains analytically intractable [64].

2.1.3.3 Particle Filter Algorithm

The particle filter persists as the foremost methodology in the literature with regards to fault prognosis due to nonlinear/non-Gaussian prediction

capabilities [64]. In addition to non-Gaussian prediction, the particle filter was originally developed for non-Gaussian state estimation for nonlinear systems. A summary of the sequential importance sampling particle filter routine for estimation and prediction is briefly presented [24, 55].

For non-Gaussian state estimation, the states can be represented as a sum of discrete weighted particles. The set of these particle locations and associated weights, $\{x_{k-1}^i, w_{k-1}^i\}$, respectively, approximate the state PDF at time, $k - 1$,

$$P(x_{k-1}|z_{1:k-1}) \approx \sum_{i=1}^{N_p} w_{k-1}^i \delta(x_{k-1} - x_{k-1}^i) \quad (2.9)$$

where N_p is the total number of particles, $\delta(\cdot)$ is the Dirac delta operator, x_{k-1}^i is location of the i 'th particle, and w_{k-1}^i is the associated particle weight.

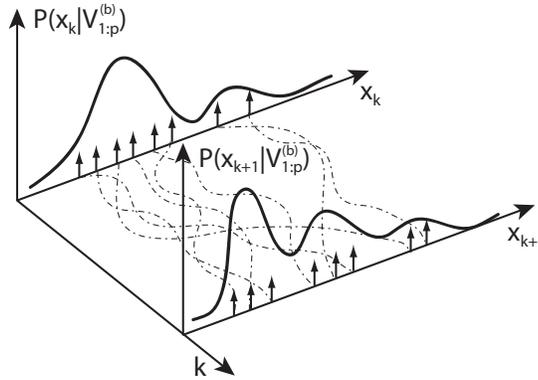


Figure 2.6: Single step illustration of particle filtering prediction.

To obtain the predicted *a priori* state estimate, the set of particles, x_{k-1}^i , is propagated through the nonlinear dynamic model, $x_k^i = f(x_{k-1}^i, u_{k-1})$. Particle propagation is illustrated in Figure 2.6. Given a new measurement,

z_k , the *a priori* estimate can be updated via weight innovation. A common assumption to simplify the particle filter implementation is to assume the importance function and prior distributions are equivalent [65]. Resultantly, the weight update given measurement innovation becomes,

$$w_k^i \propto w_{k-1}^i P(z_k | x_k^i) \quad (2.10)$$

where $P(z_k | x_k^i)$ is the likelihood function for the measurement process. The particle weights should remain normalized such that $\sum w_k^i = 1$. The posterior distribution is represented via equation (2.9) with updated weights and locations. To utilize the PF for prediction, the particles are continuously propagated through the nonlinear dynamic model. During prediction, no measurements are taken and thus the particle weights remain constant [55].

2.1.4 Metrics for Prognostic Fidelity Evaluation

To evaluate the fidelity of prognostics methodologies, numerous prediction fidelity metrics have been proposed. A full review of prognostic metrics used in academia/industry can be found in [66]. Protocol for prediction evaluation is to ensure the proposed algorithm predicts the ground truth run-time over the entire run-time of the mobile system, known as the prognostic horizon [24]. The *ground truth* (GT) run-time is the measured time of an experiment performed for prognostic algorithm verification [67]. To evaluate an algorithm over the prognostic horizon, the algorithm under study predicts RRT at specified intervals over an entire experimental study. These predictions are

evaluated against the GT run-time. Two metrics, utilized in this work for algorithm comparison, are subsequently presented.

One common accuracy-based metric, for evaluating a prognostic algorithm's prediction fidelity over the prognostic horizon, is the relative accuracy (RA) metric [68]. The relative accuracy of a prediction gives the fractional error of the predicted RT and the GT for a given prediction time. An example prognostic horizon with the RA metric is illustrated in Figure 2.7. As such, the RA at time, t_s , is given by,

$$RA(t_s) = 1 - \frac{|r_*(t_s) - r(t_s)|}{r_*(t_s)} \quad (2.11)$$

where $r_*(t_s)$ is the GT run-time and $r(t_s)$ is the predicted RT of the prognostic algorithm under evaluation. The RA metric is bounded on the interval, $RA(t_s) \in [0, 1]$, such that large prediction errors are truncated to a RA score of zero. Furthermore, the numerator of equation (2.11) is often shortened to $\Delta_{GT}(t_s) = |r_*(t_s) - r(t_s)|$ for simplicity [67].

The RA metric individually evaluates the fractional accuracy of a prognostic algorithm for a given prediction time on the prognostic horizon. Resultantly, the normalized sum of the RA scores for the entire prognostic horizon yields the cumulative relative accuracy (CRA) for a single prognostic study [24]. The CRA can be expressed as follows,

$$CRA = \frac{1}{N_p} \sum_{s=t_p}^{t_{RT}} RA(t_s) \quad (2.12)$$

where t_p is the time of the first prediction on the prognostic horizon, t_{RT} is the GT run-time, and N_p is the number of predictions over the prognostic horizon.

As the domain of the RA scores is restricted, similarly, the domain of the CRA scores is restricted to $CRA \in [0, 1]$. A perfect CRA score of unity implies the prognostic algorithm under evaluation predicts the GT run-time perfectly for each prediction made.

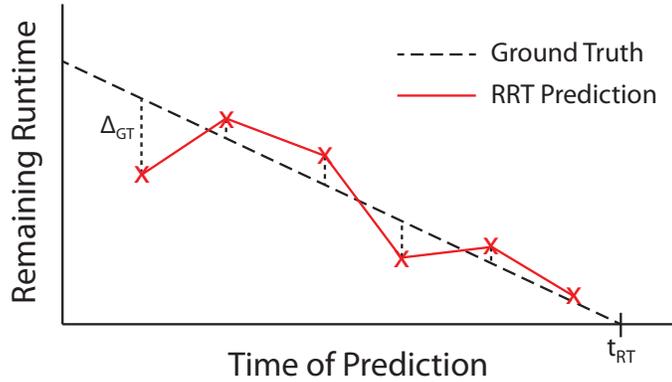


Figure 2.7: Cumulative relative accuracy prognostic evaluation metric over the prognostic horizon. Adapted from [67].

Evaluating a prognostic algorithm with solely accuracy-based metrics, such as the CRA metric, ignores the associated predicted uncertainty bounds. Resultantly, a hybrid precision/accuracy evaluation metric, the so-called the $\alpha - \lambda$ (AL) prognostic metric, was introduced in [66] to further account for prediction uncertainty. As a consequence of the dual precision/accuracy evaluation, the AL metric has seen widespread adoption for prognostic methodology evaluation in applications such as bearing fault prognosis [69], Li-ion battery state-of-health prognosis [68], and battery end-of-discharge prognosis [11]. The AL metric evaluates the total probability of an algorithm prediction which lies within a shrinking horizon, illustrated in Figure 2.8.

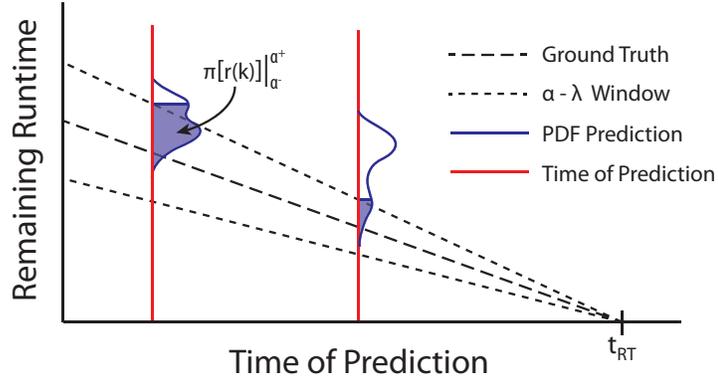


Figure 2.8: $\alpha - \lambda$ prognostic evaluation metric evaluating probabilistic prediction accuracy over the prognostic horizon. Adapted from [67].

The $\alpha - \lambda$ bounds of integration for prognostic evaluation are given by,

$$\Delta_{\alpha-\lambda}(t_s) = r_*(t_s) [1 + \alpha] \quad (2.13)$$

where $r_*(t_s)$ is the GT run-time at prediction time, t_s , and α is the window modifier. An $\alpha = 0.3$ implies a $\alpha - \lambda$ horizon of 30% with respect to the remaining RT run-time, and as such, the bounds tighten with time. Furthermore, the λ represents the normalized prognostic horizon where the GT run-time indicates $\lambda = 1$. Finally, to award the binary scores, a percentage of the predicted PDF, β must lie within the $\alpha - \lambda$ interval. Succinctly stated,

$$\int_{-\alpha}^{+\alpha} \pi [r(t_s)] \geq \beta \quad (2.14)$$

where $\pi [r(t_s)]$ is the predicted PDF at prediction time, t_s . Provided an *a priori* specified area of the predicted PDF lie within the AL bounds given by equation (2.14), the prognostic algorithm is awarded a binary score of unity for the prediction ($AL(t_s) = 1$). Otherwise, a score of zero is issued ($AL(t_s) = 0$).

Akin to the RA metric, the AL metric can be evaluated over the entire prognostic horizon for a single prognostic experiment. A normalized sum of the AL scores over the prognostic horizon yields the cumulative $\alpha - \lambda$ (CAL) metric

$$CAL = \frac{1}{N_p} \sum_{s=t_p}^{t_{RT}} AL(t_s) \quad (2.15)$$

where t_p is the time of the first prediction on the prognostic horizon, t_{RT} is the GT run-time, and N_p is the number of predictions over the prognostic horizon. As the domain of the AL scores is restricted, similarly, the domain of the CAL scores is restricted to $CAL \in [0, 1]$. A perfect CAL score of unity implies the PDF prediction of the prognostic algorithm under evaluation lies within $\alpha - \lambda$ interval over the entire prognostic horizon.

2.2 Load Characterization Methodologies

As discussed, several model-based prediction algorithms exist in the literature with respective advantages and disadvantages. Model-based prediction for battery RRT prediction remains an open problem, due to the necessity of load characterization [70]. However, in the battery RRT prediction literature, characterization of uncertain loads, imposed on a mobile system via the environment, is treated as a secondary concern to prediction. In [13], a sliding horizon average of the power loads is used for model forecasting. However, as discussed extensively, these techniques ignore power load transients that remain pivotal in the electrochemical discharge process [44]. To include load dynamics, some works [13, 29, 71, 72] have assumed full knowledge of future

power demands. Inherent to mobile systems, however, is non-deterministic nature of future loads [12, 16]. Finally, some studies have adopted stochastic load characterization using either stationary Gaussian power loads or as Markov models, to account for uncertainty in the power demands [19, 20, 73, 74]. Of the existing literature, only the studies which used jump-Markov models introduce transients in power demand [19, 20, 74]. However, in these studies, load characteristics were specified *a priori* with no data acquired online during operation. As a result, existing methods in the literature fail to characterize the variety of power loads potentially encountered by a mobile system.

2.2.1 Load Characterization and Forecasting

Generally, the load history imposed on a battery system represents a time series of a complex mobile system/environment stochastic process. For a ground robotic vehicle, a purely model-based representation of power loads would require information on upcoming terrain terremechanical properties such as terrain cohesion and internal friction angles and vehicle/terrain interaction [75]. Resultantly, a statistical characterization of the measured power loads provides a data-driven approximation to the overall process for load forecasting [76].

In the literature, numerous methodologies exist for the characterization of time series for load forecasting, e.g., regression analysis, time series modeling methods, discrete stochastic systems modeling, and artificial intelligence inspired forecasting [77, 78]. Multivariable regression, typically used in the lit-

erature for short term load forecasting [77, 79], minimizes the residual sum of squares of a function model of data. However, online regression forecasting requires *a priori* knowledge of expected loading structure (linear, polynomial, or logarithmic, etc.). Unsupervised symbolic regression, where the load structure is identified via genetic programming, has seen limited adoption [80]. However, metaheuristic optimization approaches can converge to local optima which do not accurately characterized the load process and thus limit the methodology [81]. As power loads are measured online during mobile system operation, power load data must be characterized in a self-supervised manner without requiring an operator to specify load structure [82].

Artificial intelligence (AI) inspired methodologies, such as the ANN [83], fuzzy logic [84] and support vector regression (SVR) [41] have also been used for load forecasting. As compared with regression techniques, the AI inspired algorithms allow for direct nonlinear characterization of data. Resultantly, these methods can model trends independent of direct model specification. However, ANNs are prone to model overfitting where measurement noise influences ANN trends, which results in forecasts with high variance (in the machine learning sense) [83, 85]. Unfortunately, a reduction in model-order, which reduces variance, results in attenuation of load transient characterization [79]. Furthermore, ANN and SVR require high fidelity data and the forecasting model must be retrained given newly acquired data [77, 85]. Time series modeling, via the autoregressive moving average (ARMA) scheme, also provides for statistical forecasting [76]. For example, for load forecasting, such

as power grid demand analysis which exhibits seasonality, ARMA forecasting has been shown effective [77, 86]. However, when loads remain unstructured with transient power peaks, ARMA forecasts fail to capture transient behavior for long term forecasting [77, 85].

2.2.2 Forecast Monitoring

For a stationary load process, a single accurate load characterization can provide sufficient information for all future forecasting/prediction [87]. However, power load stationarity cannot be guaranteed for a mobile system operating in an unstructured environment [21]. Resultantly, the transient load process must be recharacterized when the forecast and measurements no longer coincide [88].

The exponentially weighed moving-average (EWMA) control chart provides a method to determine when forecast residuals exceed an *a priori* specified threshold [89]. Upon exceeding the EWMA control chart threshold, the load process can be recharacterized to abate the residual error [87]. The EWMA chart statistic can be recursively computed as follows,

$$r_k = \lambda_r e_k + (1 - \lambda_r) r_{k-1}, \quad (2.16)$$

where $\lambda_r \in (0, 1)$ is the EWMA forgetting factor, and r_{k-1} is the previous EWMA statistic value. Furthermore, the difference between the one-step ahead forecast and measurement provide the error function,

$$e_k = y_k - \hat{y}_k, \quad (2.17)$$

where y_k is the measured load value and \hat{y}_k is the one-step ahead forecast made at time, $k - 1$ [87]. The control lines, which indicate the maximum and minimum healthy values of r_k , are given by,

$$C_L = \mu_e \pm L\sigma_e \sqrt{\frac{\lambda_r}{2 - \lambda_r} [1 - (1 - \lambda_r)^{2i}]} \quad (2.18)$$

where μ_e is the mean process residual, σ_e is the variance of the process residual, L is an *a priori* specified control bound (typically $L = 3$), and i is the number of samples [89].

2.3 Dynamic Battery Models

Energy-aware control and run-time prediction for battery systems both require dynamic models which account for variable current loading and the associated systems-level electrochemical phenomenon. While subtle dissimilarities in physics and performance exist between prevalent battery chemistries, the general operation and internal diffusion effects remain fundamentally comparable [7]. The following discussion notionally introduces the basics of the battery electrochemical reaction from a systems/control perspective and motivates the selection of the equivalent circuit model abstraction.

2.3.1 Systems-level Battery Process

The foundation of the electrochemical cell results from the combination of two electrodes, the anode and the cathode, which are divided by an electrolytic separator [7]. Selection of the electrolyte and electrode constituents,

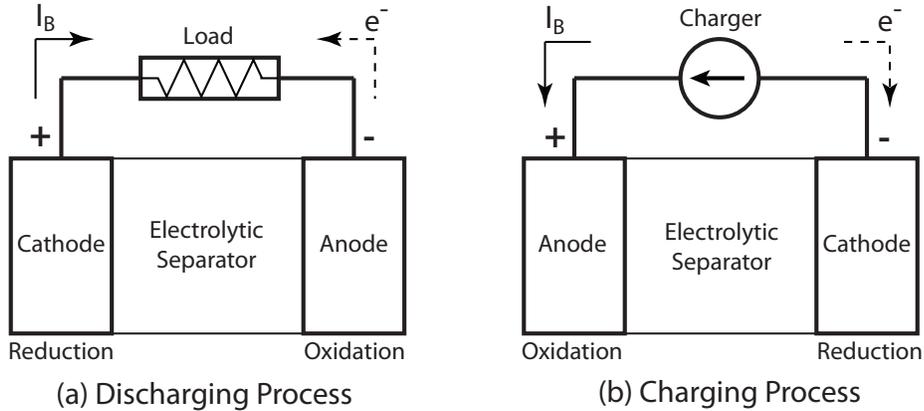


Figure 2.9: General schematic of the battery electrochemical reaction for (a) discharge and (b) charging [7].

also known as active materials, determines battery properties and specifies the cell chemistry [90]. For the NiMH battery chemistry utilized in the robot studies of this work, the electrodes are composed of nickel oxyhydroxide and a metal hydride compound for the positive and negative terminals, respectively [7]. Additionally, the electrolyte is some form of a hydrogen permeable alloy, commonly potassium hydroxide [7]. Each electrode individually forms a half-cell, which can be viewed as the solid matrix electrolytic material suspended in additional electrolyte solution [91]. Combination of two half cells forms a full electrochemical cell with the summation of each individual half-cell potential, computed via the Nernst equation, providing the overall electrochemical cell potential, or measured battery voltage [7].

Flow of internal ions dictates the process of the electrochemical system, seen in Figure 2.9. During cell discharge, the positive electrode accepts electrons, while the negative electrode loses electrons. Internally, positive ions,

or cations, flow from the anode, which oxidizes, to the cathode resulting in reduction [90]. For the NiMH chemistry, positively charged hydrogen atoms join the cathode from the anode during the discharge process [7]. Similarly for the discharge process of Li-ion chemistries, the lithium cations intercalates into the cathode electrode from the anode [92]. The electrolytic separator provides the avenue for transport diffusion and regulation of these ions between the electrodes [90]. Battery depletion results upon reaching a potential equilibrium between the two half-cells [7]. For secondary, or rechargeable, cells, the above discussion and the electrochemical reactions remain reversible. As a result, the cation diffusion direction and oxidation-reduction reaction are reversed, as seen in Figure 2.9.

Contrasting to the internal electrochemistry, the electronics/shutdown conditions of a mobile system depend on the terminal voltage of the battery cells. While cell output voltage depends on myriad factors (e.g. the electrode and electrolyte phase potentials, reaction rates, and current densities in the electrolyte [92]), the discharge curve, seen in Figure 2.10, simply relates the abstract concept of state-of-charge to open circuit battery voltage. The iconic nonlinear homeomorphism results from the change in concentration of the active solid material in the electrolyte at the anode/cathode electrodes [44].

Given a loaded battery cell, additional discharge effects must be considered to determine the terminal battery voltage, as seen in Figure 2.11. Internal resistance to load currents results in ohmic losses, which generally exhibits a linear effect. Contrastingly, activation polarization provides initial resistance

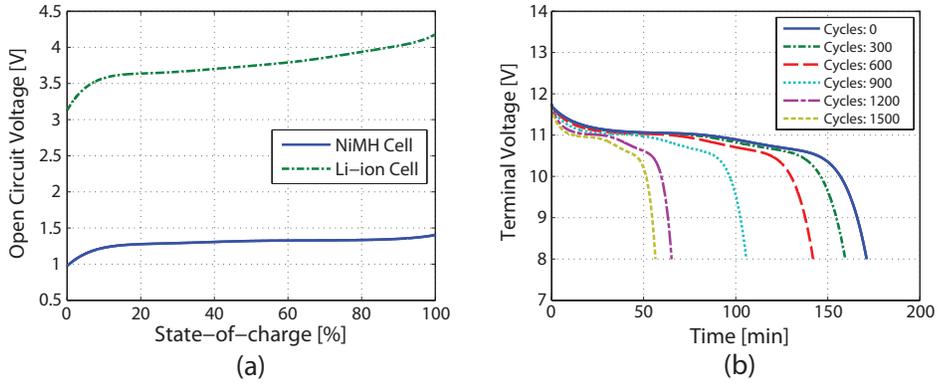


Figure 2.10: (a) Open circuit discharge curves of single cells of NiMH and Li-ion chemistries (b) Age dependency of discharge cycles illustrated for a 3000 mA-h NiMH battery pack at 1A constant load.

to diffusion of ions at the surfaces of the electrodes. Finally, a nonlinear load dependent resistance to ion flow through the electrolyte results in a concentration polarization resistance [7]. These resistances to the diffusion process provide for the recovery effect [44], where the terminal voltage of a battery will recover when unloaded. Furthermore, at higher current rates, the diffusion process losses efficiency resulting in the rate-capacity effect [7], where the apparent capacity of the battery diminishes.

2.3.2 Modeling of Battery Physics

Mathematical modeling of the electrochemical process remains an arduous process due to the subtle nuances associated with the ion reactions. Consequently, a spectrum of models exist which span the divide between model fidelity and computational efficiency [72], see Figure 2.12. High fidelity models, such as those developed by [92, 93], adhere to a first principle modeling

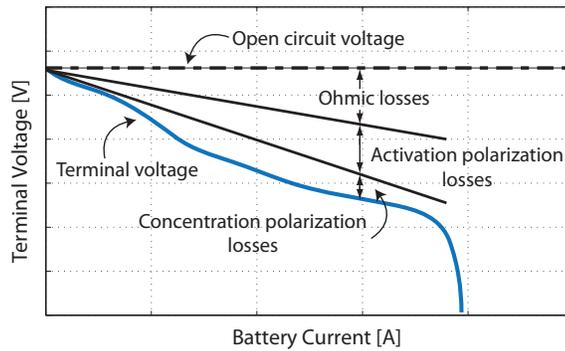


Figure 2.11: Battery cell polarization effects which link terminal voltage with load current [7].

approach and utilize porous electrode theory to model the ion diffusion in the cells. Consequentially, these physics-based models remain computationally complex preventing real time usage [93], however, averaged electrochemical models have been employed for online estimation with success [44].

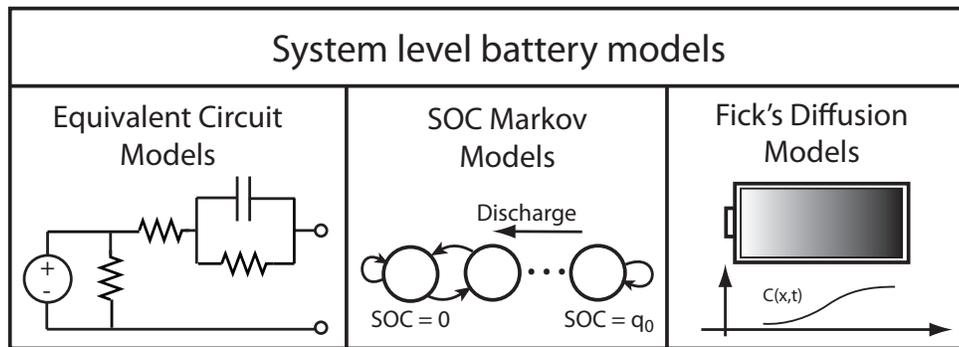


Figure 2.12: Systems level battery models in the literature appropriate for run-time prediction.

For systems-level battery modeling, physical properties of the diffusion process are selectively modeled for computational efficiency at the expense of model accuracy [94]. The most common models presented in the literature are

summarized in Figure 2.12. Inclusion of the rate-capacity and recovery effects with accurate model parameters has shown to produce models with errors of 2% for predicting terminal voltage and state-of-charge [95].

One potential systems-level representation of the electrochemical discharge process is to model the lumped diffusion process in the electrolyte through a one dimensional Fick's diffusion equation, as introduced in [71]. Under the boundary conditions and model assumptions detailed in [96], an analytical solution exists to the partial differential that can be used to solve for battery RRT under stochastic loading conditions. In controlled discharge studies, the analytical Fick's equation has been shown to accurately predict battery RRT under *a priori* known piecewise constant current loads with errors bounded by 2.7% [20]. However, these Fick's diffusion models, which have been presented in the battery RRT prediction literature [20, 29, 71, 74], have not received widespread adoption due to the lack of capabilities to incorporate temperature and aging effects into model parameters [74]. Furthermore, as the parameters are not physically motivated, parameter identification remains non-trivial [20]. Finally, the Fick's diffusion equations, as presented, discount the voltage-based failure associated with field battery operation [18].

Another systems-level representation of the battery discharge process presented in the literature uses a Markov chain to model the discharge process [97]. A chain of sequential Markov states, shown in the center of Figure 2.12, represents the battery SOC. The Markov transition probabilities represent the discharge and recovery processes. Run-time is assessed via a Markov chain

forecast realization [73]. For pulsed-load discharge tests, the SOC Markov model was shown to predict battery RRT with a bounded error of 4% [74]. While the SOC Markov chain has been presented in the research literature [73, 97–99] for small-scale wireless devices, the Markov model transition properties are not physically motivated and remain non-trivial to identify online [20].

2.3.3 Lumped Parameter Battery Model

To model the dynamic response of the battery electrochemical reactions, the equivalent circuit model utilizes the lumped parameter abstraction. In contrast to the partial differential equations that model physical ion diffusion in [93], a lumped parameter model mimics the diffusion effect via internal model current flow, and resultantly, model accuracy suffers. To model the various macroscopic electrochemical effects of the battery system, numerous ECMs have been proposed and implemented in the literature for particular applications, e.g. SOC estimation [100], battery impedance modeling [72], and health prediction [101]. A comprehensive study of the accuracy and usage of common battery ECMs can be found in [95].

The modified Thévenin ECM, illustrated in Figure 2.13, captures the critical rate-capacity and recovery macroscopic effects of the battery system and hence was selected for the prediction/control algorithms of this work. Furthermore, the Thévenin model provides reasonable model accuracy, with state and output voltage errors bounded by 2% [72] without significant computational expense. As discussed earlier, the methodology, presented in the ensuing

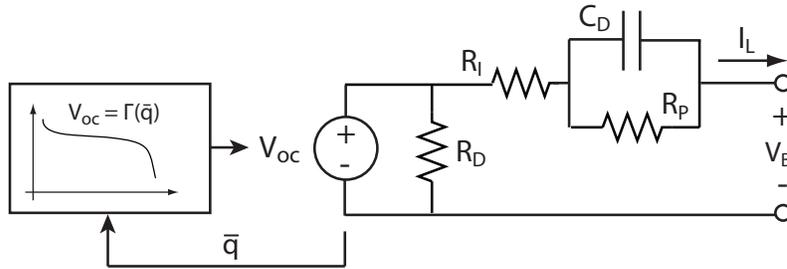


Figure 2.13: Modified Thévenin equivalent circuit model of a generic electrochemical battery system.

chapters, is not restricted to the selected model.

Remaining battery charge, q_B , forms the foundation of the Thévenin ECM. The charge indicates the remaining energy in the electrochemical system while also explicitly dictating the battery open circuit voltage through the discharge curve relationship, shown in Figure 2.13. The charge state was normalized to simplify the nonlinear state equations and to generalize the discharge curve models. Scaling the current battery charge with the maximum battery charge, q_0 , provides the normalized charge,

$$\bar{q} = q_B/q_0 \quad (2.19)$$

Additionally, the normalized charge represents the ratio of the battery charge remaining, which colloquially is known as the battery state-of-charge,

$$SOC = \bar{q} \times 100\% \quad (2.20)$$

Furthermore, the normalized charge and open circuit voltage, V_{OC} , are related through the discharge curve, expressed in this work as,

$$V_{OC} = \Gamma(\bar{q}) \quad (2.21)$$

As expected, the lumped parameters of the ECM model the physical process of ion flow. The diffusion capacitance, C_D , represents the inclination of surface ions of the electrode to initially surge from the battery provided a current load. Contrastingly, the polarization resistance, R_P , models the losses prior to the initial diffusion. Standard ohmic dissipation is captured by the general internal resistance term, R_I . The effects of R_P and R_I both contribute to the polarization effects visible in Figure 2.11. Finally, internal self-diffusion of charge, which is especially prevalent in NiCd cells [7], is given by the internal dissipation term, R_D . Combining the aforementioned terms, the nonlinear dynamic equations for the Thévenin ECM are given as follows,

$$\dot{V}_D = -\frac{1}{R_P C_D} V_D + \frac{1}{C_D} I_L \quad (2.22)$$

$$\dot{\bar{q}} = -\frac{1}{q_0 R_D} \Gamma(\bar{q}) - \frac{1}{q_0} I_L \quad (2.23)$$

where the nonlinear model states are V_D , the diffusion voltage, and \bar{q} , the normalized remaining charge. Furthermore, the model input, I_L , represents the battery load current. Relating the internal model states to the battery terminal voltage, V_B , the nonlinear model output is,

$$V_B = \Gamma(\bar{q}) - V_D - R_I I_L \quad (2.24)$$

Discounted in this model are temperature and aging effects on the battery cell. The experiments conducted for this research utilize NiMH cells with a low number of discharge cycles, and thusly temperature and aging are negligible [7]. However, extensions of the above ECM to include these effects remains feasible, see [95].

2.4 Probability of Mission Completion Assessment

Mobile systems are deployed to accomplish objectives or to complete a mission subject to finite onboard energy constraints. For example, an electric vehicle is used to complete a morning commute mission subject to the uncertainties of traffic without exhausting the onboard battery [102]. Furthermore, a teleoperated ground vehicle could be deployed to travel to a hostile building and back while conducting surveillance [12]. In each scenario, mission failure remains undesirable and results in costly/dangerous system recovery or the forfeiting of a mobile system [26]. However, reliability analysis of the mission/vehicle process can yield an estimate of probability of mission completion (PoMC) to inform the driver/supervisory control algorithm of current mission feasibility [103]. Considerable literature exists on the assessment of mobile system reliability in terms of component failure [104, 105] and vehicle mobility [106, 107]. Contrastingly, only recently have researchers addressed the need to evaluate the PoMC based on whether the necessary energy for a mission exceeds the available onboard storage [21, 108]. In the following section, existing reliability methods are discussed with a particular emphasis on evaluating PoMC for mobile vehicles.

Model-based fault prognostics methodologies for reliability/failure analysis have matured to include both online updating of probabilities and dynamic models [109–111]. Fault prognostics methods use dynamic models for statistical forecasts and assess the probability of a particular model state lying in a failure region for reliability assessment, as seen in Figure 2.14. For mobile

systems, model-based fault prognostics methodologies have been applied with a particular focus on component reliability/condition-based maintenance, such as suspension failure prognosis [112] and sensor fault prognosis [113]. Appropriately, considerable attention has been paid towards assessing the reliability of the onboard energy storage system of mobile systems. In [114], battery reliability is predicted via a model-based representation of the battery state-of-health and cell aging. Correspondingly, several other works [13, 17] present methodologies based on fault prognosis/reliability analysis to predict battery remaining run-time.

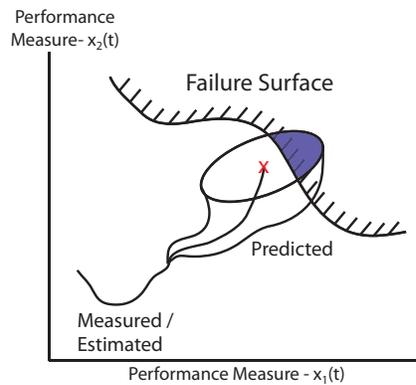


Figure 2.14: Dynamic fault prognostics and multivariate reliability analysis. [109]

Mission reliability analysis/mission planning in the literature has been address by computing the probability of the necessary mission energy exceeding the available onboard energy storage [21, 115]. In [113] and [116], the available onboard energy storage serves as a hard constraint for feasible mission planning via dynamic programming and quadratic programming, respectively.

Furthermore, in [108], a Tabu-search method is used to generate candidate mission profiles from a series of tasks to minimize energy requirements for a mobile ground vehicle. The Tabu search ensures the required mission energy never exceeds the available battery energy [108].

Finally, in [21], a Bayesian regression methodology is used to forecast mission energy requirements. A reliability integral is used to compute, $P(E_m(k) > E_{th})$, where E_m is the predicted mission energy and E_{th} is the energy failure threshold. However, as discussed, for battery powered systems, the rate-capacity effect of the electrochemical discharge reaction invalidates direct energy considerations for PoMC estimation [74]. As a result of the rate-capacity effect, two missions with identical energy requirements, but different time/power demands, can result with two different final battery SOC [117]. Resultantly, existing techniques proposed in the literature do not accurately assess the PoMC of a mobile system with finite onboard energy storage.

2.5 Energy-aware Control Schemes

Energy-aware control algorithms, as presented in the literature, attempt to account for constraints imposed by finite energy storage systems [118] when issuing control commands. In the literature, a range of energy-aware schemes exist which can be generally segregated as power load distribution optimization and input/task regulation. Resulting from the explicit constraints in limited energy source applications, researchers have advocated for such schemes as model predictive control (MPC), dynamic programming,

or constrained global optimization methodologies which directly account for state restrictions. Subsequently, supervisory control has found wide adoption in applications such as chemical process regulation [119], robot trajectory control [25], and power electronics [120] amongst other implementations.

Task-scheduling energy-aware applications have *a priori* knowledge of the necessary tasks to complete and the available onboard energy. Supervisory control, for these applications, determines the optimal scheduling of tasks conditioned on the constraints of the system. For instance, in [10], the discrete operation modes of a remote network system are chosen to minimize charge depletion of a battery system subject to latency constraints. In essence, the algorithm exploits the charge recovery effect [74] of the battery system providing for deeper cell discharge [37]. Others have employed similar optimization routines with variations on the objective function for energy-aware applications such as single battery powered CMOS circuits [121] and parallel battery packs on electric vehicles [122]. A limitation of these task-scheduling schemes is the requirement for *a priori* knowledge of tasks/power loads such that the optimization routine can schedule discharge and recovery times [123].

Contrastingly, power load distribution control takes a reversed approach. Rather than scheduling tasks conditioned on energy storage limitations, the control scheme schedules the usage pattern of the stored energy given a stream of tasks/loads. For instance in [124], supervisory control dictates the usage of both the battery and fuel cells in a hybrid vehicle, conditioned on the constraints of battery SOC. Similar supervisory load allocation controllers have

been developed for plug-in hybrid electric vehicles [22]. For non-dispatchable energy generation (e.g. wind energy), optimal allocation (and turbine control) of generated energy to the grid and storage systems provides for near-optimal energy generation/storage [125]. As such, supervisory control schemes can control both the storage mechanism and the optimal load profile. However, as discussed, small scale mobile systems commonly have only a single energy storage mechanism [8]. Resultantly, for small unmanned ground vehicles with battery energy storage, the load distribution techniques remain invalid.

Although supervisory control has been utilized for the control and mission planning of mobile systems given uncertainty, limited work has been done to incorporate energy storage constraints. In [126], a differentially driven robot followed a path trajectory via a model predictive framework with Smith predictors to account for time delays. Additionally, in [25], researchers implemented robust MPC on a robot with unknown slip for trajectory tracking. In each of the aforementioned works, the optimization includes nonholonomic constraints of limited lateral slip, actuator saturation, and robot velocity limits. Only recently have constraints on the energy storage mechanism been considered for mission planning via MPC. In [106], graph theory optimization is employed to minimize energy drawn from the battery system while traversing between waypoints. However, minimization of energy consumption does not prevent mobile system shutdown [127, 128]. For small unmanned ground vehicles, unexpected vehicle shutdown commonly results from transient power demands [129].

2.6 Summary

As an extensive review of the literature has indicated, existing methods for mobile battery system prediction and control ignore the effects of transient loads. Power load transients, when imposed on battery systems, activate systems-level electrochemical effects, such as the rate-capacity and recovery effects. Furthermore, algorithms, in the literature, that do incorporate transient load structures require *a priori* load characterization, which restricts applications to systems with known loads. However, in areas of continued growth, such as electric vehicles [130] and ground robotics [131], power demands remain *a priori* unknown and contain significant transients due to stochastic environmental demands [12]. Consequentially, considerable work remains to extend current battery prediction/control capabilities to include online characterized transient load structures.

Additionally, with the continued increase in deployable computational power, work in the literature points towards online implementation of prognostics routines. As such, computationally complex algorithms, such as the particle filter, are being used for online run-time prediction. A particular interest to this work is the increased application of the particle filter to battery remaining run-time prediction [15, 132]. In the existing literature, the particle filter run-time predictions have been made offline with simple discharge tests. Consequentially, considerable work remains to extend the theoretical particle filter predictor to online applications such as ground robot operating in an unstructured environment.

Existing conjectures in the literature have proposed the direct evaluation of mission energy versus onboard energy to determine the probability of mission successes. For battery operated systems, which account for most mobile robotic systems [133] and 3% of the US automotive market as of 2012 (hybrid/electric vehicles) [134], the electrochemical rate-capacity effect clearly invalidates these claims. Resultantly, the online identification of the correlation between mission energy and onboard energy remains an open problem.

Finally, the methods presented in the literature for energy-aware control have not been extended to include mobile systems, such as ground robots, which have a single onboard energy source. Existing controllers presented in the literature optimize the energy allocation between multiple energy sources, such as ICEs and batteries in hybrid vehicles. Other controllers optimize a finite set of tasks to ensure completion prior to energy exhaustion. Resultantly, considerable work remains to design a controller which attempts to maintain mobile system operation when subject to potentially infinite tasks and transient power loads.

Chapter 3

Online Stochastic Load Characterization

Systems operating with unstructured loads inherently exhibit stochasticity and probabilistic jumps/transients. This chapter introduces the mathematical nomenclature required for stochastic system characterization and the Gaussian Mixture/Jump-Markov (GMJM) algorithm. Systems of interest in this dissertation, as discussed in Section 1.2, operate in environments without deterministic loading conditions, and consequently model-based predictions must account for this load uncertainty online, see Figure 3.1, without *a priori* information [17]. Furthermore, uncertain transient load characterization allows for prediction utilizing nonlinear dynamic model effects, such as the battery rate-capacity and recovery effects [44]. Finally, as discussed in Section 1.2.2, the validation of this work was performed with multiple unmanned ground vehicle systems. While the discussion remains tailored to these particular UGV platforms, the following discussion and methodology is generalizable to a wide range of systems.

Successful traversal of uncertain terrain by a tele-operated/fully-autonomous ground robot requires myriad sensors/actuators, each with varied power demands, for obstacle detection, surveillance, and locomotion [4, 12, 82].

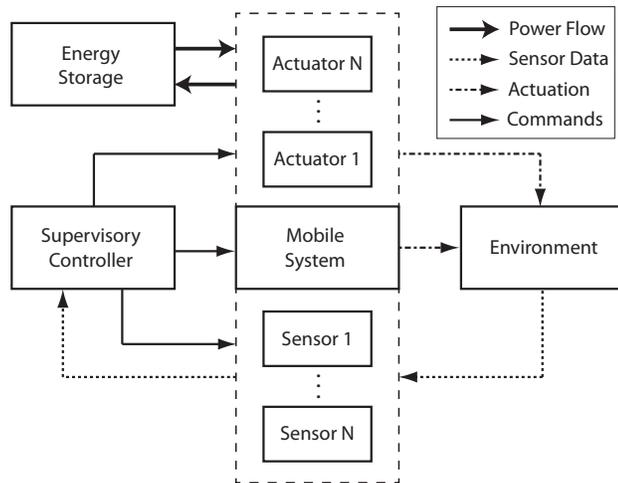


Figure 3.1: Environmental and mission profile (specified by the teleoperator/supervisory control) effects through periodic sensor/actuation usage on system energy storage.

As an example, power-hungry sensors, such as light detection and ranging (LIDAR) or visible/infrared spectrum cameras, may remain powered for full time navigation/terrain inspection [82], or could be employed cyclically for periodic surveillance depending on current environmental demands [135]. Furthermore, steady state locomotion demands vary tremendously according to the terrain terramechanical properties, vehicle command speed, and terrain inclination [21]. Consequently, loads are: *a priori* unknown, inherently transient in nature, and likely non-Gaussian.

As a consequence of the *a priori* unknown transient nature of unstructured loads and the necessity for statistical jump characterization for dynamic predictions, an online methodology for load characterization was developed. Transient loads have historically been represented via the jump-Markov chain

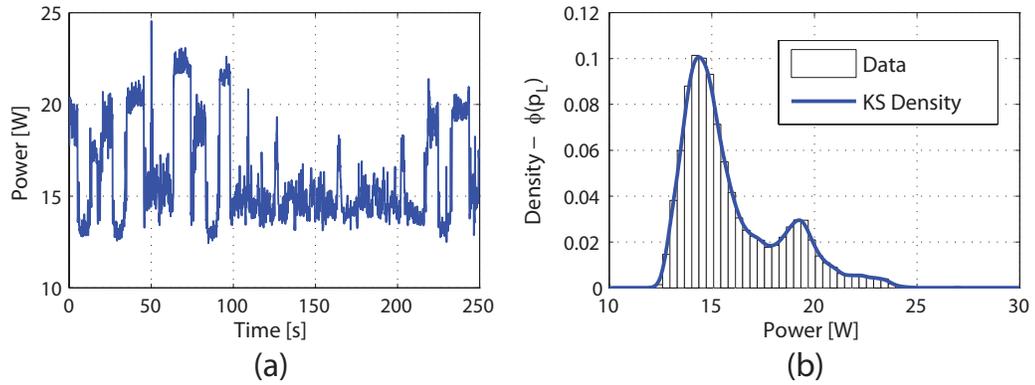


Figure 3.2: (a) Example transient power loads imposed on ground robot battery system. (b) Non-gaussian kernel density fit of battery power loads.

process, which models the probability of transitioning between load states (or Markov states), with many reported successes [16, 19]. However, these existing works assume full *a priori* knowledge of the transition properties of the loads and provide no methodology for online learning of load demands. The following methodology employs unsupervised clustering algorithms which determine the structure of the jump-Markov chain without human input. As a result, a system encountering unstructured loads can “learn” power demands and ultimately utilize the characterized loads for prediction and control.

The remaining portion of this chapter is devoted to the mathematical representation and online characterization of uncertain transient loads. Section 3.1 introduces the jump-Markov process for transient load representation. Subsequently, Section 3.2 introduces the Gaussian mixture clustering algorithm which performs online clustering of measured load data. For online self-supervised implementation of the methodology, the Akaike information

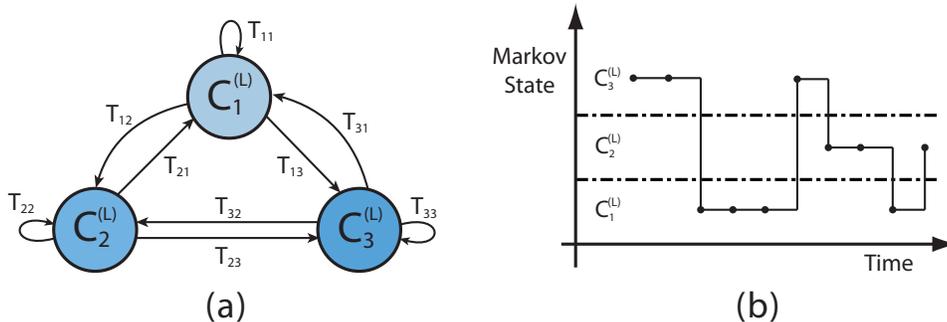


Figure 3.3: (a) Jump-Markov transition characteristics example between different load states. (b) Single realization of a Markov load trajectory.

decision metric is employed, which is discussed in Section 3.3. Finally, the integrated algorithm for load forecasting is presented in Section 3.4.

3.1 Jump Markov Process Modeling

Resulting from the uncertain transient behavior demonstrated for systems operating with unstructured loads, the jump-Markov process was employed to mathematically represent the statistics of transient load behavior. Akin to standard Markov chain theory, the jump-Markov process represents the probability of transitioning from a current state to any other possible model state, see Figure 3.3.

3.1.1 Markov Chain Theory

Each model state has a given probability of realization, typically expressed in vector notation as $P(X_n)$. As time progresses, the probability of the current load state will conditionally depend on all previous states

$P(X_k) = P(X_k|X_k - 1)$, requiring knowledge of all existing states. Resulting from the causality enforced by reality, however, the physical load demands abide by the Markov property [16, 77]. The Markov property states that the conditional probability distribution of the Markov chain simplifies to rely on only the previous probability [136]. As a result, the conditional probability simplifies to the following,

$$P(X_{n+1} = x|X_n = x_n) = P(X_{n+1} = x|X_n = x_n, \dots, X_1 = x_1) \quad (3.1)$$

where X_1, \dots, X_n are from stochastic (loading) process, $X(t)$, which is defined on the countable set S . The set, S , contains all possible load states and defines the Markov chain order. Furthermore, assuming that the loading process is generated by some unstructured environment which can be fully characterized, the probabilities of transition, as illustrated in Figure 3.3(a), will remain constant for the system throughout time. Consequentially, the following assumption implies that the unstructured terrain remains isotropic in terms of the transient load statistics [17]. As a result, however, the loading process remains time-independent and the load Markov chain similarly remains a stationary process [136], or generally,

$$P(X_{n+1} = b|X_n = a) = P(X_n = b|X_{n-1} = a). \quad (3.2)$$

where a and b are two particular load states. Resulting from the stationary assumption of equation (3.2) and the Markov property of equation (3.1), the Markov chain load process can be uniquely defined by a constant square matrix of size, S . This constant matrix, known as the transition matrix, describes the

probability of transitioning from each prior state in set S to the subsequent state [136]. Mathematically, the transition matrix is defined as follows,

$$T_{ij} = P(X_{n+1} = j | X_n = i), \quad (3.3)$$

where the $\sum_{j=1}^N T_{ij} = 1$ constraint ensures a total probability for state transition of unity for each given state and $i, j \in S$. Upon the identification of transition characteristics of the Markov process, the computation of the conditional probabilities requires only simple matrix multiplication as follows,

$$P(X_{n+1}) = P(X_n)^T T_{ij}. \quad (3.4)$$

3.1.2 Jump-Markov Process Realization

For realization, the jump-Markov process prior state is uniquely known (the input vector has the following deterministic form, $X_n = [1 \ 0 \ 0]^T$, for a three state jump-Markov process where the prior location is the first state). Furthermore, the realization of the jump-Markov process will provide the subsequent Markov state, which will either remain the same or transition (or jump) to another possible state. Mathematically, the realization remains conditionally dependent on the previous state realization, or succinctly,

$$x_t = X_t(\omega) | x_{t-1} = X_{t-1}(\omega) \quad (3.5)$$

where ω indicates the random process sample space, $\omega \in S$, that contains the set of all possible outcomes, x_t and x_{t-1} are the current and previous realization respectively, X_t is the current Markov state, and X_{t-1} is the known

prior Markov state. Weighted random number generation adequately realizes equation (3.5) for our purposes [48].

3.1.3 Characterization of a Markov Process

Provided with a sequence of load transitions, a jump-Markov chain that describes the transitions statistics can be ascertained via maximum likelihood estimation (MLE). Direct measurement of power loads on the battery bus results in a sequence of power measurements. These measurements must first be clustered into respective discrete load states (Markov states) which is addressed in Section 3.2. These measured load states are expressed as, $c^{(L)} = \{c_1^{(L)}, c_2^{(L)}, \dots, c_n^{(L)}\}$, where the elements $c_i^{(L)}$ are the power cluster data belonging to the set, $c^{(L)} \in 1, 2, N$ which defines all possible Markov states. MLE optimizes the likelihood function provided the aforementioned load sequence for a statistical model to determine the model parameters [85].

Assuming independence of observation, the joint probability of observing the cluster data, given by $c^{(L)}$, can be expressed via the union of conditional probabilities of observing all individual clusters in the sequence, or succinctly stated,

$$P(X_n = c_n^{(L)}) = P(X_1 = c_1^{(L)}) \prod_{t=2}^n P(X_t = c_t^{(L)} | X_{t-1} = c_{t-1}^{(L)}) \quad (3.6)$$

Resulting from the Markov assumption stated in equation (3.3), the probabilities of transitioning remain constant for identical state transitions.

Consequently, the likelihood function can be expressed as,

$$L(\theta) = P(X_1 = c_1^{(L)}) \prod_{i=1}^k \prod_{j=1}^k \theta_{ij}^{n_{ij}} \quad (3.7)$$

where θ_{ij} and n_{ij} are the unknown state transition probabilities and number of transitions from state i to state j respectively and k is the *a priori* specified number of Markov states. Furthermore, the maximum probability of transitioning from one state to any other cannot exceed unity ($\sum_j \theta_{ij} = 1$), and thusly, constraints must be enforced for MLE [85]. Taking the logarithm of equation (3.7) and enforcing the probability constraint via a Lagrange multiplier yields the following MLE problem,

$$L(\theta) = \log P(X_1 = c_1^{(L)}) + \sum_{i,j} n_{ij} \log \theta_{ij} + \sum_{i=1}^k \lambda_i (1 - \sum_j \theta_{ij}) \quad (3.8)$$

Maximization of the log-likelihood function with respect to the model parameters, θ , yields the following method for identifying the transition characteristics,

$$\theta_{ij} = T_{ij} = \frac{n_{ij}}{\sum_{j=1}^k n_{ij}} \quad (3.9)$$

3.2 Gaussian Mixture Clustering

During constant system operation or steady state periods, power demands have been shown to exhibit Gaussianity for robotic ground vehicles [122]. In-house experimental power load analysis, shown in Figure 3.4, further corroborated the above claim. For twenty-two discharge studies, the skewness

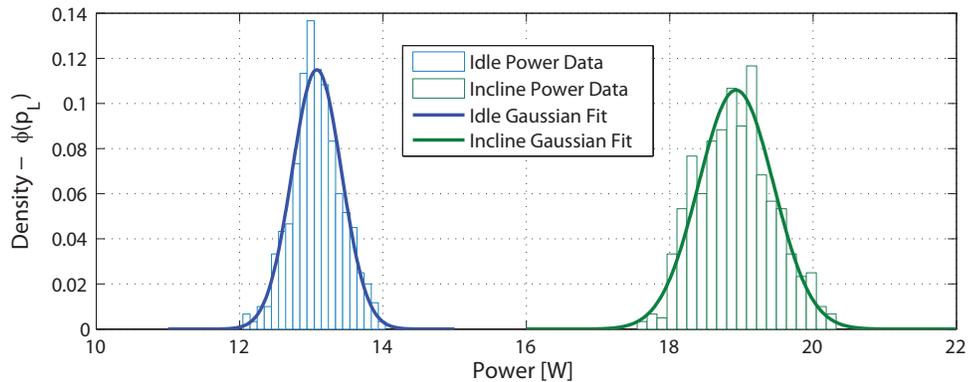


Figure 3.4: Illustration of Gaussianity of power loads during steady state operation of a ground vehicle in two modes: Idling and locomotion at 0.25 m/s on 25 degree inclined tile terrain.

and kurtosis of the idle power loads were -0.098 ± 0.05 and 0.064 ± 0.13 , respectively. For the forward inclined power loads, the skewness and kurtosis were 0.029 ± 0.06 and -0.098 ± 0.09 , respectively. As a result, *a priori* characterization of all expected steady state operation modes of the ground vehicle would provide a Markov chain load characterization [19]. However, *a priori* characterization of loads potentially discounts varied driving styles between vehicle operators, unexpected/uncharacterized terrains, and extreme operation requirements [17]. Consequently, the robotic system should identify the Markov load structure online without requirements for *a priori* power measurements. Online identification of the Markov structure requires a self-supervised clustering methodology, such that the robotic system can determine the structure of loads without human intervention. Such a technique requires online clustering capabilities and a decision metric to determine the number of clusters [137].

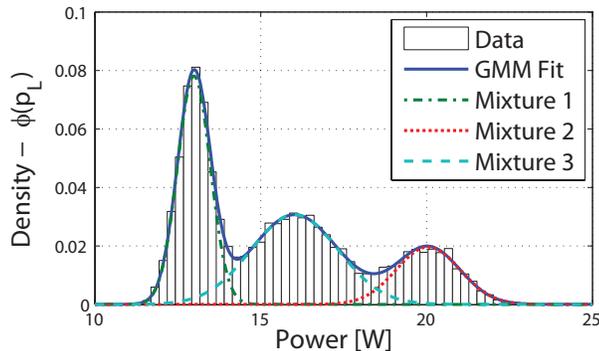


Figure 3.5: Gaussian mixture model approximation to a non-Gaussian load distribution.

While numerous clustering methodologies exist in the literature, such as hierarchical and k-means clustering [85], the Gaussianity of the steady state loads suggests that Gaussian mixture clustering could identify load clusters directly/autonomously. The Gaussian mixture clustering algorithm takes a vector of measured power load data, $p^{(L)} = \{p_1^{(L)}, p_2^{(L)}, \dots, p_n^{(L)}\}$, that is potentially non-Gaussian as a whole, and determines Gaussian sub-models (a Gaussian mixture model). This clustering process is notionally illustrated in Figure 3.5 and 3.6.

The probability density function for the entire load sequence can be represented via weighted sums of the Gaussian sub-models, as follows,

$$P_{\theta}(x) = \sum_{k=1}^M \alpha_k \phi(x|\mu_k, \sigma_k^2) \quad (3.10)$$

where the model parameters to be found are $\hat{\theta} = [\hat{\alpha}_k, \hat{\mu}_k, \hat{\sigma}_k^2]$, and the distributions of $\phi(x|\mu_k, \sigma_k^2)$ are normally distributed. The α_k values are mixture weights of the Gaussian distributions that satisfy the normalizing condition,

$\sum_{k=1}^M \alpha_k = 1$. Furthermore, M dictates the number of clusters. As with finding Markov chain parameters, optimization of the mixture model likelihood function provides the statistical model parameters, as follows,

$$\theta_{MLE} \in \arg \max_{\theta \in \Theta} P_{\theta}(X|\theta) \quad (3.11)$$

where $P_{\theta}(X|\hat{\theta})$ is the likelihood function and Θ is the parametric space. For ease of numerical implementation and the nonlinearity of the GMM likelihood function, the MLE for a GMM is generally computed via the expectation-maximization (EM) algorithm [138]. The EM algorithm, which is discussed theoretically in detail for GMMs in Appendix C, iteratively calculates successively higher likelihood parameters until reaching specified convergence properties [85]. In the expectation step, the likelihood function is computed via equation (3.12) with the current parameter estimates.

$$P(C_m|x_i) = \frac{\hat{\alpha}_m \phi(x_i|\hat{\theta}_t)}{\sum_{m=1}^M \hat{\alpha}_m \phi(x_i|\hat{\theta}_t)} \quad (3.12)$$

Subsequently, during the maximization step, the parameters are updated via equations (3.13), (3.14), and (3.15).

$$\alpha_k = \frac{1}{N} \sum_{j=1}^N P(C_k|x_j, \theta_t) \quad (3.13)$$

N is the total number of data point and C_k are the current clusters. In a similar fashion, the update equations for μ and σ can be computed as follows,

$$\mu_k = \frac{\sum_{j=1}^N x_j P(C_k|x_j, \theta_t)}{\sum_{j=1}^N P(C_k|x_j, \theta_t)} \quad (3.14)$$

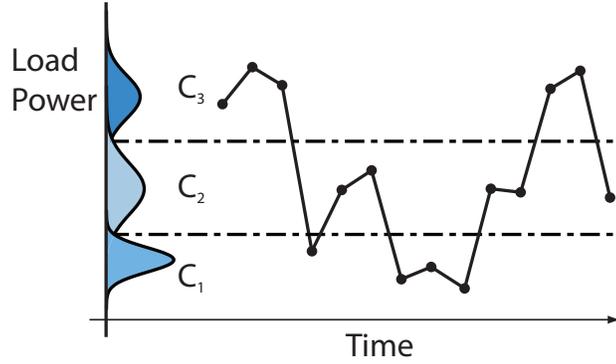


Figure 3.6: Clustering of measured load data into local Gaussian models.

$$\sigma_k = \frac{\sum_{j=1}^N \left[(x_j - \mu_k)(x_j - \mu_k)^T \right] P(C_k | x_j, \theta_t)}{\sum_{j=1}^N P(C_k | x_j, \theta_t)} \quad (3.15)$$

The nonlinearity of the GMM MLE optimization problem implies possible existence of local minima at which the algorithm could erroneously converge. Reinitialization of the EM algorithm initial conditions, $\hat{\theta}_0$, can heuristically help mitigate poor routine convergence in accordance with [85]. For the implementation of GMM clustering, in this work, the EM algorithm is reinitialized five times and run with identical convergence properties, i.e. maximum number of iterations and step size minimum threshold for the likelihood function. The five optimized likelihood functions are compared and the parameters that demonstrate the overall maximum likelihood are selected as optimal.

Each cluster contains information on the mean, variance and associated model weight. As a result, when load forecasting, a load realization in cluster one will exhibit the Gaussian characteristics given by cluster one. For this

work, the Gaussian time series characteristics are assumed to be a white noise process [76]. Transition characteristics between load clusters is found via the equation (3.9), which requires the load data to be clustered, see Figure 3.6. Clustering of the load data vector, $p^{(L)} = \{p_1^{(L)}, p_2^{(L)}, \dots, p_n^{(L)}\}$, with the GMM routine provides the clustered load vector, $c^{(L)} = \{c_1^{(L)}, c_2^{(L)}, \dots, c_n^{(L)}\}$. Given the GMM model parameters and the load vector, $p^{(L)}$, the probabilities of each load belonging to each cluster are computed via the posterior probability,

$$P(C_m^{(L)}|p_i^{(L)}) = \frac{\hat{\alpha}_m \phi(p_i^{(L)}|\hat{\mu}_m, \hat{\sigma}_m^2)}{\sum_{m=1}^M \hat{\alpha}_m \phi(p_i^{(L)}|\hat{\mu}_m, \hat{\sigma}_m^2)}, \quad (3.16)$$

where $C_m^{(L)}$ are the individual mixture models. Equation 3.16 only soft clusters power loads. Commonly, the maximum a posterior (MAP) classification methodology is employed [137] to select the cluster with the highest probability to classify each load value, see Figure 3.6. Provided the posterior probability in equation (3.16), the MAP classification for a particular load, $p_i^{(L)}$, is given by,

$$c_i^{(L)} = \arg \max_{m \in \{1, 2, \dots, M\}} P(C_m^{(L)}|p_i^{(L)}), \quad (3.17)$$

where $c_i^{(L)}$ are the classified load points. With the load cluster sequence, $c^{(L)}$, MLE of the Markov transition characteristics becomes feasible.

3.3 Unsupervised Model Selection

For unsupervised identification of the GMJM model structure, the number of Gaussian clusters must be determined without *a priori* specification.

Computationally, cluster number determination remains a NP hard problem and typically heuristic implementations require iterative EM algorithm fitting for different cluster numbers [85]. Relative fit metrics provide for an assessment of tradeoff between model complexity and model fidelity. Strictly increasing the number of mixtures of the GMM will always result in greater maximum likelihood functions, and thus increasingly complex models must be penalized.

For relative model comparison, several decision criteria exist including the Akaike information criterion (AIC), the Bayesian information criterion (BIC), the minimum description length (MDL) and cross-validation (CV), to name a few [102]. The AIC metric was chosen for this work due to the reported successes of unsupervised identification of cluster numbers with the GMM methodology [139], however other relative metrics are anticipated to perform similarly. The AIC metric is typically expressed as follows,

$$AIC = 2k - 2 \ln Q \quad (3.18)$$

where k is the total number of model parameters in θ and Q is the maximized likelihood function which results from the EM algorithm during GMM parameter optimization. The number of parameters, k , for the GMM is given by $3M - 1$ where M is the number of clusters [140]. Given the power load vector and parameters in θ , the maximized likelihood can be computed as follows,

$$Q = - \sum_{k=1}^M \alpha_k \phi(p^{(L)} | \mu_k, \sigma_k^2) \quad (3.19)$$

As can be seen in equation (3.18), minimization of the AIC penalizes progressively more complex models while recompensing more accurate models.

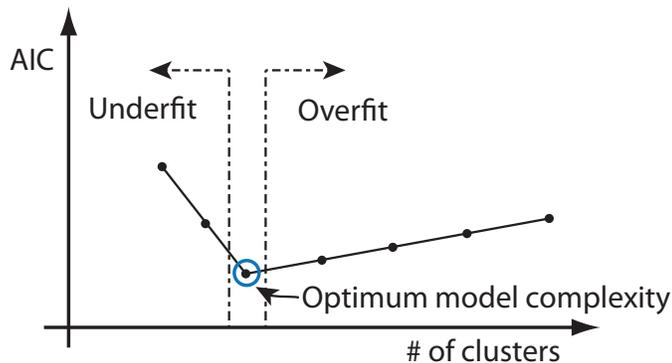


Figure 3.7: Unsupervised model identification via Akaike information criterion.

Global minimization of the AIC provides the GMM fit with the optimal number of mixture models [137]. As discussed above and illustrated in Figure 3.7, sequential fitting of GMMs and evaluating the AIC provides a straightforward method for unsupervised identification of cluster number.

However, heuristic constraints on the fitting routine must be further imposed. In this implementation, the total cluster space is limited to a maximum of 15 clusters resulting from the increasing computational complexity associated with high cluster numbers [85]. Upon reaching 15 clusters without finding a local AIC minimum, the iterative routine will end and select the 15 cluster fit. This restriction is justified for the ground robot verification experiments, in this work, due to the relative low number of power load clusters [17]. Future implementations of this methodology should remain cognizant to this limiting assumption and perform some power analysis prior to restricting cluster number.

3.4 Integrated Algorithm

Integration of the above techniques provides a methodology for online characterization of transient power loads referred from hence forth as the Gaussian mixture jump-Markov (GMJM) algorithm. For a given load sequence, Gaussian mixture clustering coupled with AIC can identify the number of load regions and the Gaussian parameters that characterize the load regions. Upon MAP clustering of the load vector into the identified load regions, $c^{(L)}$, the transient characteristics of the load sequence can be identified via a MLE fit of a jump-Markov process. Resultantly, full characterization of a transient load process requires only the Gaussian mixture information and the jump-Markov transition matrix. Graphically, the overall GMJM methodology is given in Figure 3.8.

Realization of a characterized GMJM process for prediction requires a dual realization of both the Markov chain and the associated Gaussian process. Given the previous GMJM cluster, the probability at time $k + 1$ for each cluster can be found via matrix multiplication with the transition matrix, T_{ij} , see equation 3.4. A weighted number realization of the posterior probability, $P(X_{k+1})$, yields the predicted cluster, \hat{c}_{k+1} . Secondly, with the predicted cluster realization, a Gaussian random variable realization from GMM cluster \hat{c}_{k+1} results in a predicted power load from the GMJM characterization. As the clusters are normally distributed, a normal random number generator is sufficient to generate the predicted load, \hat{p}_{k+1} . Sequential realizations imply that the predicted cluster for $k + 1$ becomes the prior cluster to predicted for

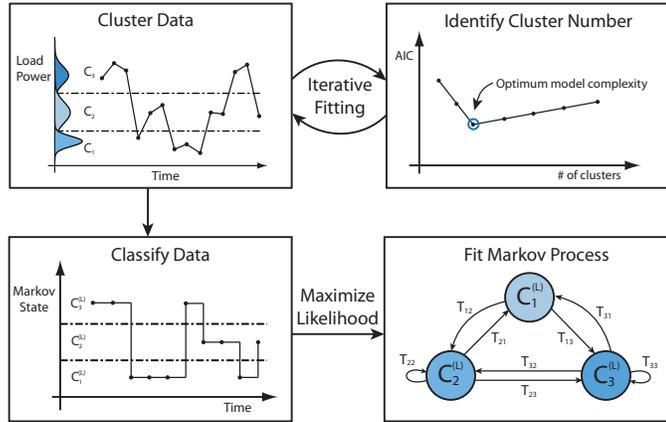


Figure 3.8: Gaussian mixture and jump-Markov transient load characterization methodology.

time $k + 2$.

3.4.1 Ground Vehicle Load Characterization

To illustrate the GMJM methodology, the technique is applied to battery load data acquired from a small ground robotic vehicle operating on an unstructured terrain. The experimental setup used to acquire the transient power demands is discussed in further detail in Section 4.4. For this particular data set, the robot does not utilize sensor cycling, and load transients result from only terrain/motor command differences. Terrain includes loose gravel, inclined linoleum tile and rough surface elevated obstacles. Each terrain and idling, as expected, requires different power demands. A sample of measured transient data is illustrated in Figure 3.9(a).

Overall, 64.19 minutes of power data was collected for this particular test prior to system shutdown due to power system enforced voltage shutdown.

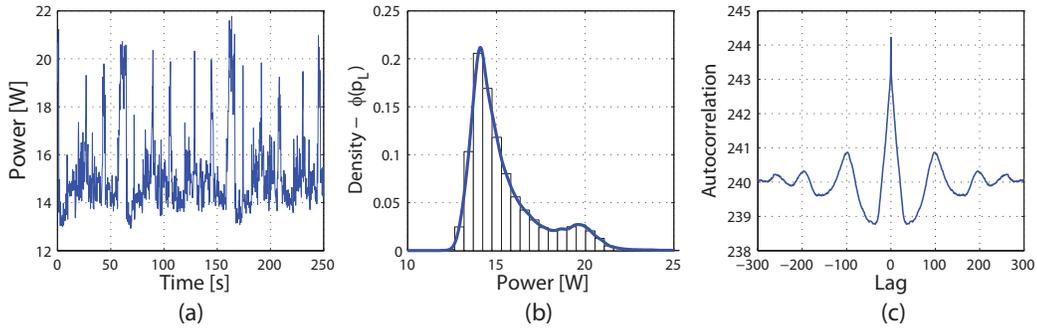


Figure 3.9: (a) Sample load power data and current terrain. (b) Histogram of entire power load vector (c) Autocorrelation of power load.

A histogram of the entire sequence of load data is given in Figure 3.9(b) which illustrates the non-Gaussianity of the overall load sequence imposed on the robotic system. As expected, the autocorrelation of power load data illustrates the lack of independence of the load data. The measured autocorrelation function follows the trend of the autocorrelation of a rounded square wave, suggesting jump-transients [76].

For the following load characterization demonstration, a transient power vector of 5000 samples, or approximately 10 minutes of data, is used for GMJM algorithm. The data logging system on-board the mobile robot records the battery load current and terminal voltage at a rate of 8 Hz. Cross-validation of model integrity given a truncated sample horizon is proven in Section 3.4.2. The GMJM algorithm begins by iteratively fitting GMMs and evaluating the AIC to determine cluster number in a self-supervised fashion. The AIC metric for this load sequence example is given by Figure 3.10(a). Upon identification of the cluster number, the GMM is found. For this particular example, the

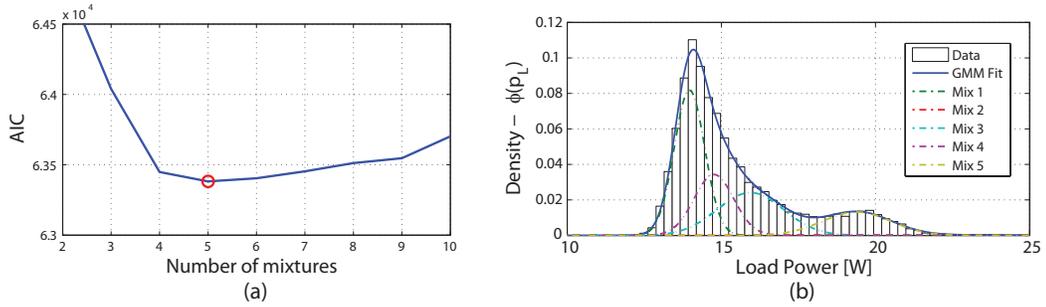


Figure 3.10: (a) Akaike information criterion identifying five Gaussian clusters for example unstructured load characterization. (b) Resulting Gaussian mixture model provided AIC cluster number identification. Cluster two, nearly indiscernible, appears to model high powered infrequent transients.

overall GMM and the Gaussian components are illustrated in Figure 3.10(b).

Given the identification of Gaussian regions, the load vector, $p^{(L)}$, can be clustered via MAP classification to provide the clustered load vector, $c^{(L)}$. This classified vector provides for the identification of the transient characteristics of the load vector. An overlay of actual power data and classified data is shown in Figure 3.11(a). Provided the classification of power data, MLE yields the jump-Markov properties of the GMJM model. Figure 3.11(b) depicts the transition matrix probabilities for the jump-Markov system.

The preceding analysis provides a full characterization of the transient jump characteristics of a measured power load without *a priori* specification of model order. For the model-based particle filter algorithm, which will be discussed in detail in Section 4.1, the Markov transition matrix and Gaussian cluster information is used to realize individual transient loads. Contrasting with typical load averaging methodologies, which propagate averaged load

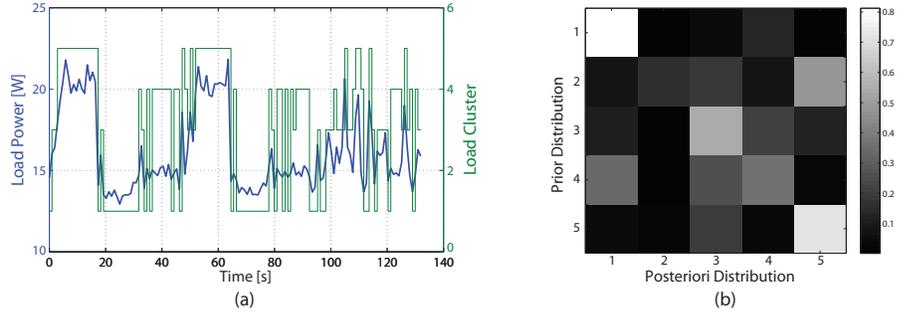


Figure 3.11: (a) Classification of power data into cluster data. (b) Jump-Markov transition characteristics.

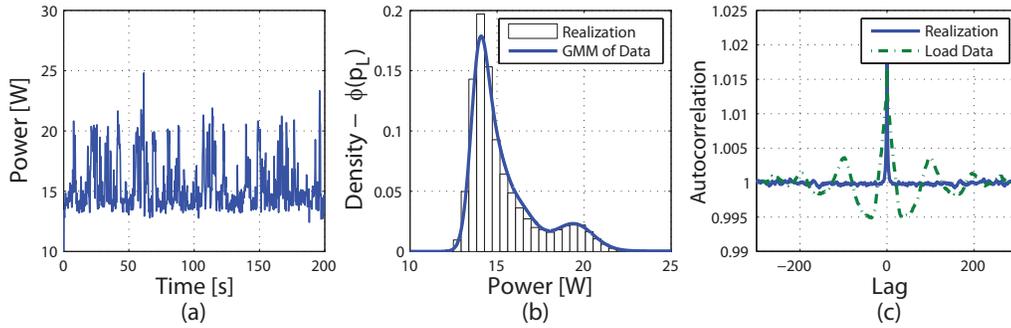


Figure 3.12: (a) Individual realization of the characterized GMJM load process. Comparison of actual load process versus a single realization of the GMJM load process as a (b) kernel density estimation function comparison and (c) autocorrelation functions.

models, the particle filter allows for transient load propagation [17].

3.4.2 Sample horizon cross-validation

A trade-off exists between computational time for GMJM characterization and the size of the load history, or training set size. Furthermore, large training sets result in overtraining, whereby a model lacks generalizability and results in high prediction variance [85]. However, small training sets poten-

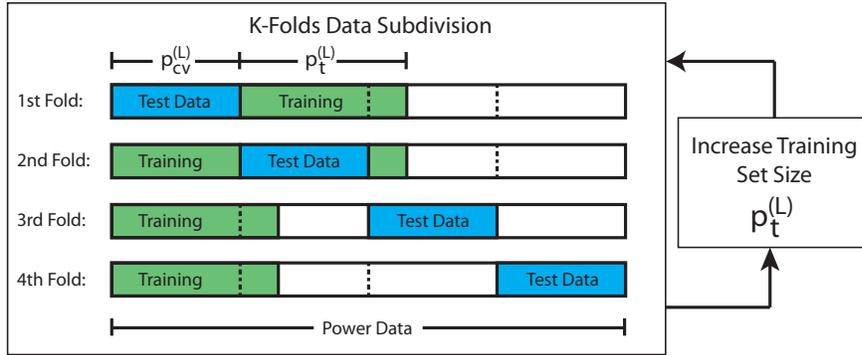


Figure 3.13: Data subdivision for k-folds analysis with training data and test data for likelihood calculation, and for each k-folds CV, the training data size is increased. [85]

tially lack sufficient information and yield models with large prediction bias. As a result, the GMJM model should be trained with a training set vector of sufficient length to capture load transient information, but sufficiently short to avoid non-stationary changes in load demands.

Model cross validation allows for analysis for the generalizability of a statistical model [85]. To account for training set variation, K-fold cross validation utilizes random subsets of the entire data vector to both train and validated the statistical model, see Figure 3.13. For the GMJM process, a subset of the entire data set, $p_t^{(L)}$, is used to characterize a GMJM model. Using subsets of the remaining data, $p_{cv}^{(L)}$, the model fidelity can be evaluated through calculation of the cross validation likelihood functions given by the GMM and Markov chain likelihood functions [23].

Provided the example load process used Section 3.4.1, an iterative comparison of training set size was conducted. For a sample window ranging from

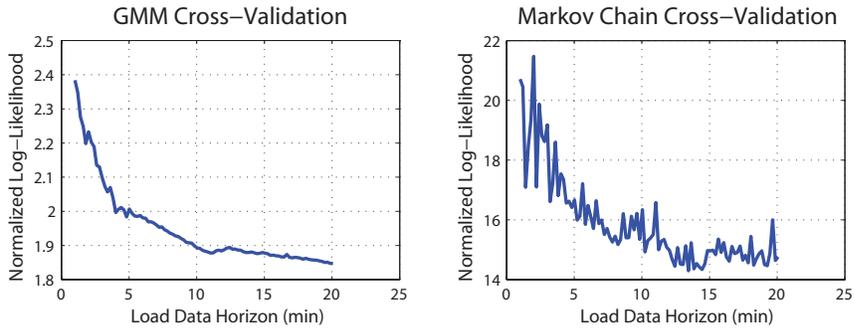


Figure 3.14: Results of K-fold cross validation study, where both GMM and Markov models exhibit minimal improvement beyond a sample size of ten minutes.

60 seconds to 20 minutes in increments of 12 seconds (100 new samples), a K-folds routine was run to identify fidelity of both the GMM and Markov models for the validation set, $p_{cv}^{(L)}$. For each cross validation, fifteen K-fold sets were used. Increasing the sample window used for GMJM model characterization is shown to have minimal effect beyond 10 minutes, as seen in Figure 3.14. Consequently, all future analysis in this body of work utilizes the 10 minute sample horizon for model training. Furthermore, to ensure consistency of prediction/load characterization, loads are recharacterized prior to each prediction. This contrasts with potential EWMA control chart monitoring discussed in Section 2.2.2. Online adaptive forecast monitoring is recommended for future works.

Chapter 4

Battery Remaining Run-time Prediction

Online prediction of battery remaining run-time (RRT) for systems operating in stochastic environments requires model-based prediction. As reviewed in Section 2.1, existing battery run-time prediction schemes fail to incorporate either the transient prediction requirements or the electrochemical battery dynamics. Load realizations from the GMJM scheme address the need for online transient load characterization. In this chapter, both the nonlinear battery model, from Section 2.3.3, and the GMJM load characterization scheme, from Chapter 3, are united with an augmented particle filter methodology for battery RRT prediction. This integrated scheme is henceforth referred to as the GMJM/PF prediction methodology.

The ensuing sections are arranged to first introduce the particle filter and culminate with two experimental studies that demonstrate the predictor efficacy. Section 2.1.3.1 introduces the mathematical notation of the particle filter and details the necessary considerations for the battery RRT prediction application. The prognostic metrics, introduced in Section 2.1.4, are used to validate prediction accuracy. With these metrics, the results of a Monte Carlo simulation study are presented in Section 4.3. These Monte Carlo simula-

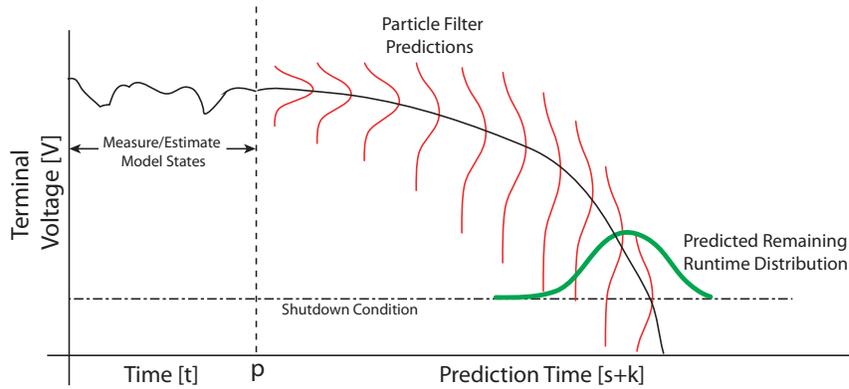


Figure 4.1: Particle filter forecasting of non-Gaussian states for battery RRT prediction.

tions demonstrate loading conditions under which the PF methodology outperforms existing battery RRT prediction schemes. Finally in Section 4.4, the GMJM/PF prediction methodology is validated with two experimental studies.

4.1 Implementing Sequential Monte Carlo

While the particle filter has been previously used for battery run-time prediction [13, 15, 55, 132], the PF routine presented in the following section incorporates novel GMJM load realizations for transient prediction. Furthermore, the PF implementation presented in this work utilizes the EKF battery state estimates from Section A.3 to mitigate additional computational complexity [141]. The following section details the modifications necessary to numerically approximate the Chapman-Kolmogorov equation for battery RRT prediction with the GMJM process.

As introduced in Section 2.1.3.1, the state probability density functions for each discrete time step can be approximated via a set of weighted values. The summation of the particles and their corresponding weights yields the approximation for the initial probability density function, as follows,

$$P(x_p|V_{1:p}^{(b)}) \approx \sum_{i=1}^N w_p^{(i)} \delta(x_p - x_p^{(i)}) \quad (4.1)$$

where $w_p^{(i)}$ are the particle weights, $x_p^{(i)}$ are the particle values, $\delta(x_p - x_p^{(i)})$ is the dirac delta function of each particle, N is the number of particles and p is the time of prediction initiation. As the ECM battery model presented in Section 2.3.3 requires only two independent states, the individual state particles are given as, $x_p^{(i)} = \begin{bmatrix} V_{D_p}^{(i)} & \bar{q}_p^{(i)} \end{bmatrix}^T$. The sum of the weighted values effectively models the uncertainty associated with model states, unstructured model inputs, and stochastic outputs. There are many varieties of the particle filter methodology described in the literature. For prediction and fault prognostics, the sequential importance resampling (SIR) algorithm, shown in Figure 4.2, is used [13, 17].

Prior to prediction, the current battery model state estimate $(\hat{x}_{k|k}, P_{k|k})$ from equations (A.12) and (A.13) must be approximated via weighted particles. As the EKF returns a Gaussian state estimate for the diffusion voltage and normalized charge, the $2 \times N$ particles are initialized uniformly over the 99% confidence interval of the estimates. After generation of the initial $x_p^{(i)}$ values, the likelihood of measurement of each particle instance dictates the associated particle weight [64]. As the measurement noise is assumed to be Gaussian, the

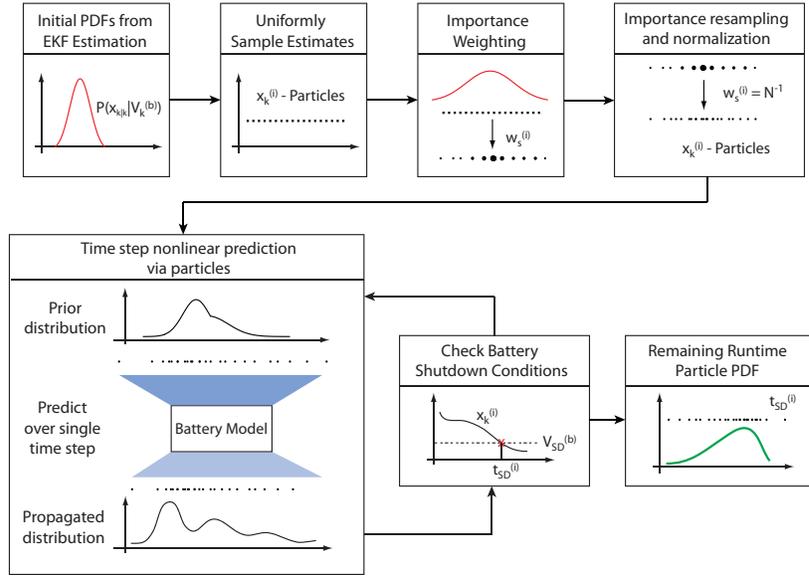


Figure 4.2: Sequential importance resampling implementation of the particle filter for model-based battery RRT prediction.

measurement likelihood for each particle realization is given as,

$$P(V_p^{(b)} | x_p^{(i)}) = (2\pi R)^{-1/2} \exp \left[-\frac{1}{2R} (V_p^{(b)} - g(x_p^{(i)}))^2 \right] \quad (4.2)$$

where R is the measurement noise variance and $g(\cdot)$ is the battery measurement process given by equation (2.24).

To improve computation efficiency of the particle filter algorithm, the weight calculation is simplified by assuming the proposal distribution is the prior distribution [142]. Furthermore, to avoid particle degeneracy during initialization, sufficient ($N > 50$) particles are realized. With the aforementioned assumptions, the initial particle weights are given by the normalized likelihood,

$$w_p^{(i)} = \frac{P(V_p^{(b)} | x_p^{(i)})}{\sum_{i=1}^N P(V_p^{(b)} | x_p^{(i)})}. \quad (4.3)$$

The initially derived weights are used for weighted resampling of the particles such that the weights become uniform and the resampled particles approach the initial state estimates, $(\hat{x}_{k|k}, P_{k|k})$. Weight resampling can also be bypassed provided the computational ability to generate normally distributed random variable realizations [64].

The weights remain constant for the prediction interval from the time of the final measurement, p , to the final prediction time $p + n$. Resultantly, the particle filter one-step prediction only requires propagation of each particle through the dynamic battery model, shown in Figure 2.6. Mathematically, particle propagation is given as follows,

$$x_{k+1}^{(i)} = f \left[x_k^{(i)}, \widehat{p}_k^{(i)} \right] \quad (4.4)$$

where $f \left[x_k^{(i)}, \widehat{p}_k^{(i)} \right]$ is the nonlinear battery dynamic model given by equations (4.14) and (4.15) and $\widehat{p}_k^{(i)}$ a unique realization of the GMJM load process for each particle. The summation of the particles and their associated weights yields the predicted state distributions,

$$P(x_{k+1}|V_p^{(b)}) \approx \sum_{i=1}^N w_p^{(i)} f \left[x_k^{(i)}, \widehat{p}_k^{(i)} \right]. \quad (4.5)$$

Likewise, the predicted probability density function for the battery terminal voltage can be computed. The predicted state particles and GMJM load process propagate through the battery output equation,

$$P(V_{k+1}^{(b)}|V_p^{(b)}) \approx \sum_{i=1}^N w_p^{(i)} g \left[x_{k+1}^{(i)}, \widehat{p}_{k+1}^{(i)} \right] \quad (4.6)$$

4.1.1 Particle-based Evaluation of Remaining Run-time

The prediction scheme outline in the previous section provides for long-term forecasting of battery states. To evaluate the RRT of the battery system, the probability of the battery terminal voltage exceeding the shutdown voltage (the hazard zone) is computed during each prediction interval. After the dynamic propagation of the particles, the approximate battery terminal voltage PDF, $P(V_{k+1}^{(b)}|V_p^{(b)})$, is compared to the shutdown voltage distribution, shown in Figure 4.3. The probability of shutdown at prediction step $p + n$ is given by the reliability integral,

$$P(V_{SD} > V^{(b)})_{p+n} = \int_{-\infty}^{\infty} \Phi_{SD}(V) P(V_{p+n}^{(b)}) dV \quad (4.7)$$

where $\Phi_{SD}(V)$ is the cumulative distribution function of the shutdown voltage and $P(V_{p+n}^{(b)})$ is the predicted battery terminal voltage PDF given by equation (4.6).

As the voltage and shutdown distributions are non-Gaussian, the integral in equation (4.7) must be computed numerically [109]. Kernel density estimation was used to smooth the discrete particle estimates to a continuous numerical PDF approximation [62]. With the numerical approximations of the voltage PDF and the shutdown CDF, Riemann sum integration is used to numerically compute the reliability integral of equation (4.7).

The shutdown probability is evaluated iteratively through each prediction step, and resultantly, the CDF of shutdown time is numerically constructed, as shown in Figure 4.3. Finally, shutdown cumulative probability

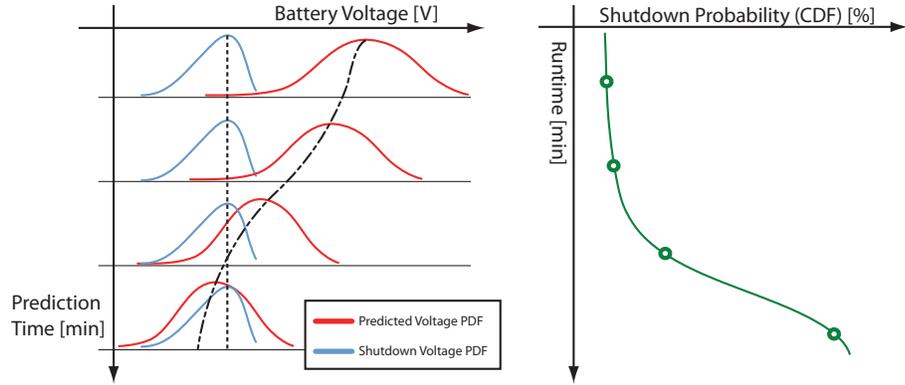


Figure 4.3: (a) Cartoon depiction of the predicted battery terminal voltage distribution via the particle filter as compared to the shutdown voltage distribution. (b) Sequential generation of the RRT cumulative distribution function.

over 99% serves as the exit condition for the particle filter prediction routine. The resulting prediction is an empirical CDF which represents the probability of shutdown at a given instance in time, as follows

$$\hat{P}(t_{s+n}) = P(V_{SD} > V_{s+n}^{(b)}) \quad (4.8)$$

4.1.2 Scalar Run-time Prediction

While the GMJM/PF algorithm returns CDF run-time prediction, mobile system operators or other algorithms require a scalar prediction of run-time. Using the CDF run-time prediction from (4.8), the median occurs at the prediction time where $\hat{P}(t_{s+n}) = 0.5$. The predicted time at this value is the median RRT, $\hat{\mu}_{RRT} = t_k \left[\hat{P}(t_{s+n}) = 0.5 \right]$ [143]. The summation of the predicted RRT and the current mission time, $t^{(c)}$, yields a prediction for the

overall battery run-time.

$$\hat{\mu}_{RT} = t^{(c)} + t_k \left[\hat{P}(t_{s+n}) = 1/2 \right] \quad (4.9)$$

Additionally, although the GMJM/PF predicted CDFs can be non-Gaussian, the RRT prediction can be approximated as normal if necessary for implementation with other methods. In particular, such an approximation is necessary for the mission probability scheme introduced in Chapter 5. Akin to the calculation of equation (4.9), the variance of the RRT prediction can be approximated as follows,

$$\hat{\sigma}_{RT}^2 = \max \left\{ t_k \left[\hat{P}(t_{s+n}) = -\sigma \right]^2, t_k \left[\hat{P}(t_{s+n}) = +\sigma \right]^2 \right\} \quad (4.10)$$

where σ is the standard normal standard deviation [143]. The maximum of the upper and lower confidence intervals ensure the variance prediction is conservative and does not under-approximate the uncertainty.

4.2 Battery Model for Prediction

For long term forecasts, the battery model must be reformulated with power load inputs to ensure conservation of energy [144]. The load current in equation (2.22) actually requires knowledge of the load impedance, or information on both current draw and terminal voltage. As a result, the battery system prescribes the voltage, or effort, into the system (typically into a battery protective circuit or power electronics system), and the load dictates the current, or flow. Voltage variation over time is undesirable for normal operation of DC electronics [112], and accordingly, a DC bus line is regulated via

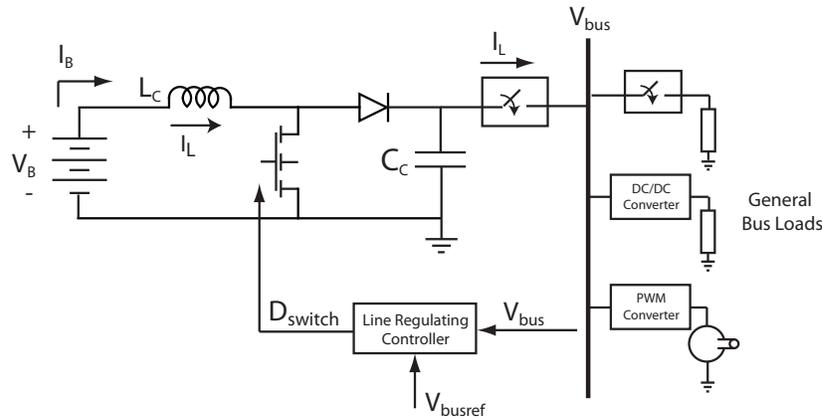


Figure 4.4: Representative implementation of voltage bus regulation with battery system.

switch-based power electronic systems [145]. A typical implementation of a line regulating DC-DC boost converter, coupled with point-of-load converters in cascade form, is illustrated in Figure 4.4 [146].

In this work, the switch-power electronics are assumed to be fast regulated with dynamics on the order of 10 microseconds or faster, and thusly the power electronics can be modeled as algebraic relationships [145]. Additionally, the power converter is viewed as a lossless process (given the utilization of only switch components and inductors/capacitors) [112]. As a result of the voltage regulation and lossless converters, a constant current load on the bus line will appear to the battery as a constant power load, see equation (4.11). As the battery voltage decays, the power converter will draw higher currents to compensate and to allow the external load to draw a constant current, but

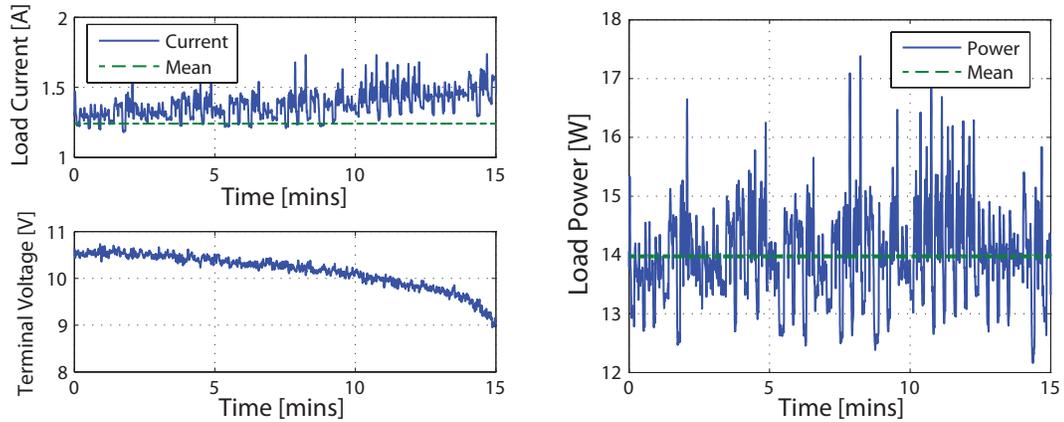


Figure 4.5: Battery load discharge test with robot vehicle drawing constant current loads on the voltage bus, resulting in a constant power load on the battery system.

conservation of energy dictates constant power draw,

$$P_L = P_M = I_M V_{bus}, \quad (4.11)$$

where P_L is the load power on the battery pack, P_M is the load power on the bus, V_{bus} is the fast regulated bus voltage which remains constant, and I_M is a bus current load. Since the loads appear as constant power loads, utilization of currents for forecasting will result in a statistical bias, as is illustrated in Figure 4.5.

Resultantly, the battery dynamics, given by equations (2.22) and (2.24), must be rederived to account for power load inputs. As such, the prior equations become subject to the following constraint,

$$I_L = \frac{P_L}{V_B} \quad (4.12)$$

where V_B is the battery terminal voltage, given by equation (2.24). Factoring in the explicit power load constraint and performing algebraic manipulations, the battery terminal output voltage given an input power load is given as follows,

$$V_B = \frac{1}{2} [\Gamma(\bar{q}) - V_D] + \frac{1}{2} \sqrt{V_D^2 - 2V_D\Gamma(\bar{q}) + \Gamma(\bar{q})^2 - 4R_I P_L} \quad (4.13)$$

Incorporation of the power constraint into the battery model produces a nonlinear output equation with a non-affine input function, P_L . Substitution of $I_L = P_L/V_B$ into the ECM state equations with the solution for V_B given in equation (4.13) yields the following augmented state equations,

$$\dot{V}_D = -\frac{1}{R_P C_D} V_D + \frac{2}{C_D} \Upsilon(P_L, V_D, \bar{q}) P_L \quad (4.14)$$

$$\dot{\bar{q}} = -\frac{1}{q_0 R_D} \Gamma(\bar{q}) + \frac{2}{q_0} \Upsilon(P_L, V_D, \bar{q}) P_L \quad (4.15)$$

where the nonlinear state dependent function, $\Upsilon(P_L, V_D, \bar{q})$, is given as follows,

$$\Upsilon(P_L, V_D, \bar{q}) = \left[\Gamma(\bar{q}) - V_D + \sqrt{V_D^2 - 2V_D\Gamma(\bar{q}) + \Gamma(\bar{q})^2 - 4R_I P_L} \right]^{-1} \quad (4.16)$$

As a result of the power constraint on the battery model, the dynamic equations now exhibit regions of infeasibility in the input function, P_L . The input load power remains feasible provided membership in the subsequent subset of \mathbb{R} given by, $P_L \in [-\infty, 1/4R_I (V_D^2 - 2V_D\Gamma(\bar{q}) + \Gamma(\bar{q})^2)]$. Intuitively, the above constrained input set implies the instantaneous load power drawn for the battery system is limited.

4.3 Load Considerations for Algorithm Selection

A drawback of the proposed GMJM/PF methodology is the computational complexity of the algorithm. The particle filter computational complexity is given by $O(N)$ with N as the number of particles [54] whereas the EKF model-based predictor is $O(N^2)$ where N is the number of model states [147]. As the PF requires considerable particle coverage for each state [147], the PF computational complexity quickly surpasses the EKF.

Consequentially, utilization of the GMJM/PF scheme over the EKF should yield significant improvement in prediction fidelity to justify the implementation costs. As the GMJM/PF methodology was developed to overcome existing prediction limitations associated with transient uncertain loads, a Monte Carlo simulation study was conducted to explore the relationship between load cluster separation (and hence the magnitude of transient jumps) and the prediction fidelity of the GMJM/PF, EKF, modified Peukert's and linear regression predictions. The other prediction methodologies were introduced in Section 2.1.

4.3.1 Second-order Markov Load Simplification

To model the transient power loads on a battery system, a two-state Markov jump chain was utilized, see Figure 4.6. This simple load process was chosen for analysis as the steady state probabilities and transition rate characteristics of the second-order Markov process can be specified directly [148].

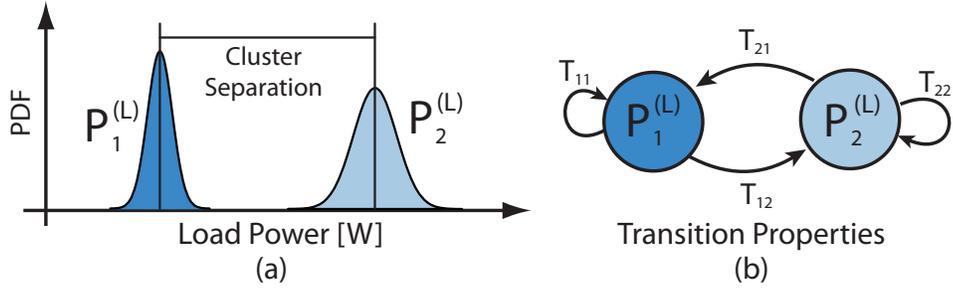


Figure 4.6: Simple loading process for algorithm evaluation.

Provided the steady state cluster probabilities and transition rates of the second-order process, the Markov transition matrix can be explicitly found,

$$T_{ij} = \begin{bmatrix} 1 - \pi_2 - \lambda\pi_2 & \pi_2 + \lambda\pi_2 \\ \lambda\pi_2 - \pi_2 - \lambda + 1 & \pi_2 + \lambda - \lambda\pi_2 \end{bmatrix} \quad (4.17)$$

where π_2 is the steady state probability of load two and λ is the transition rate [148]. As the probabilities of each load sum to unity, $\pi_1 = 1 - \pi_2$. Furthermore, the steady state total load is held constant such that only transient characteristics change. As such, to ensure a constant steady state power load, the following relationship relates the two load values,

$$p_2^{(L)} = \pi_2^{-1} \left[\bar{p}^{(L)} - \pi_1 p_1^{(L)} \right] \quad (4.18)$$

where $\bar{p}^{(L)}$ is the specified power load average and $p_1^{(L)}$ and $p_2^{(L)}$ are the resulting Markov state load values. Cluster separation is defined as the difference between the power clusters,

$$\Delta p^{(L)} = \left| p_2^{(L)} - p_1^{(L)} \right|. \quad (4.19)$$

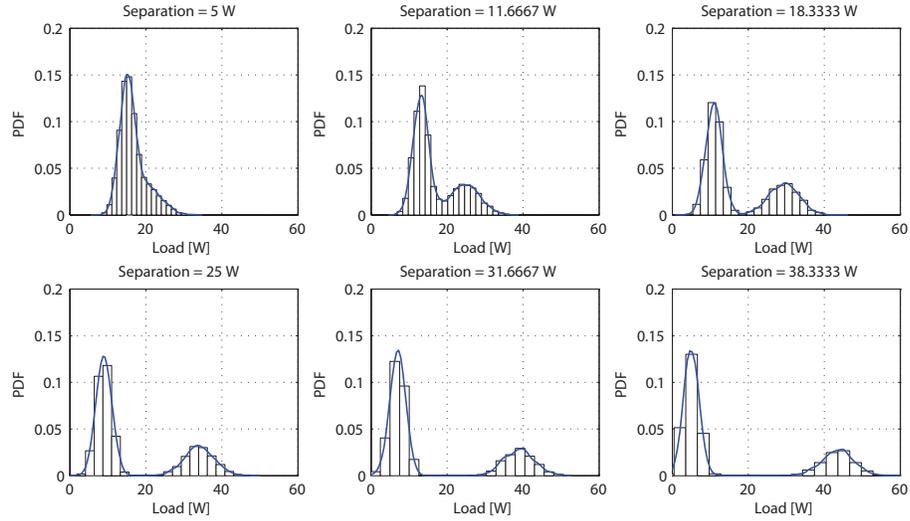


Figure 4.7: Basic load structure realizations with ranging cluster separation.

4.3.2 Prediction Fidelity and Load Cluster Separation

To explore the relationship between the size of transient jumps in a loading process (cluster separation) and the prediction fidelity, a Monte Carlo simulation study was conducted. In the simulation study, a battery model, described by the ECM detailed in Section 4.2, is subjected to a stochastic load and each prediction methodology is employed over the entire prognostic horizon, introduced in Section 2.1.4.

To ensure approximate run-time consistency, the overall load process was ensured to require identical steady state load power of $\bar{p}^{(L)} = 16.5$ W. The battery model was subjected to second-order power load transients with cluster separation, $\Delta p^{(L)}$, ranging from 5 W to 47 W. Probability density functions of the loading process as applied during the Monte Carlo simulations are illustrated in Figure 4.7.

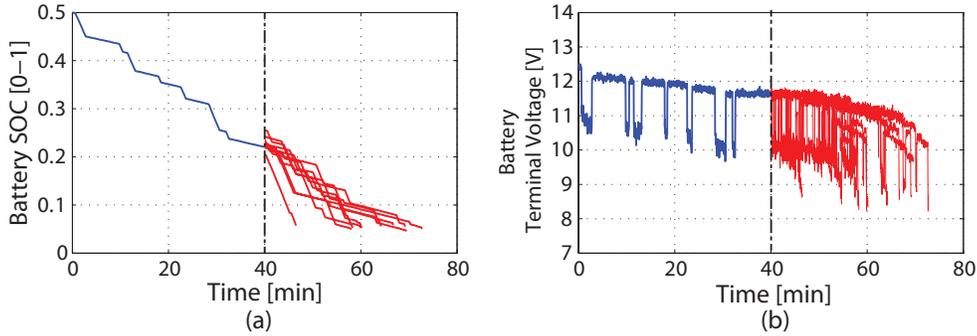


Figure 4.8: GMJM/PF prediction realization with ten particle paths with a BLS of 40W cluster separation.

The prediction methodologies are evaluated over the realizations of the prognostic horizons at intervals of 15 minutes to 45 minutes with increments of 5 minutes. A realization of the GMJM/PF prediction scheme at 40 minutes with $\Delta p^{(L)} = 40$ W is illustrated in Figure 4.8. Furthermore, the discharge simulation with the same load process is repeated 25 times for statistical significance. Finally, over each prognostic horizon realization, the CRA and $\alpha - \lambda$ metrics are evaluated. The prognostic parameters for $\alpha - \lambda$ were selected as $\beta = 0.35$ and $\lambda = 0.1$ corresponding with [66].

As is clearly visible from the CRA results displayed in Figure 4.9, cluster separation influences the fidelity of battery RRT predictions. The error bars associated with each prediction methodology indicate the standard deviation of the 150 simulations at each particular $\Delta p^{(L)}$. In terms of the CRA metric, the GMJM/PF predictor results in higher fidelity predictions at all power separation magnitudes. Furthermore, the GMJM/PF predictor significantly outperforms the other investigated methodologies particularly with $\Delta p^{(L)} > 20$

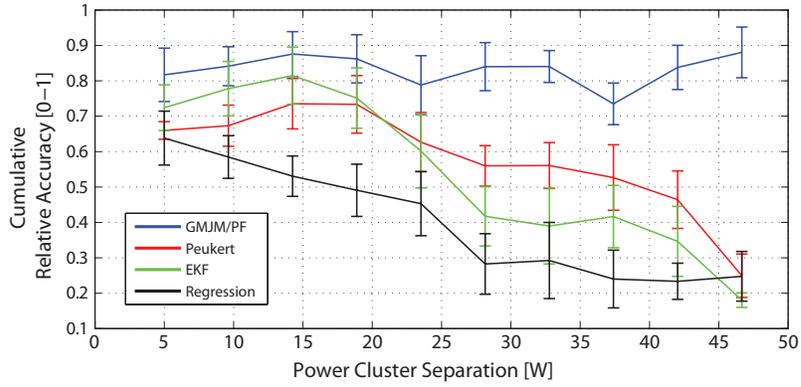


Figure 4.9: Monte Carlo CRA prediction fidelity as dependent on $\Delta p^{(L)}$.

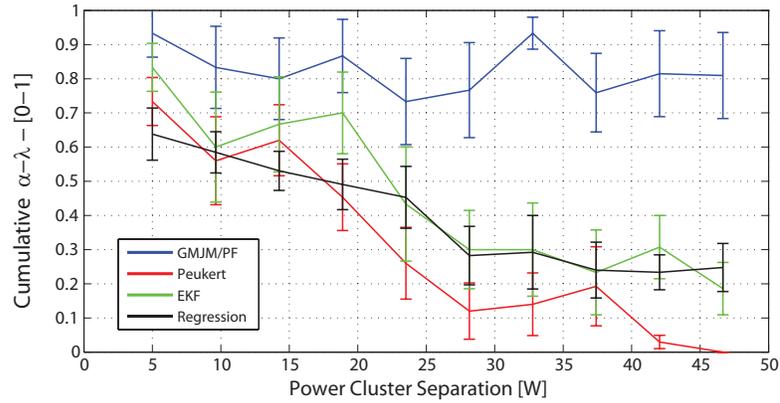


Figure 4.10: Monte Carlo $\alpha - \lambda$ prediction fidelity as dependent on $\Delta p^{(L)}$.

W.

To further collaborate with the observed CRA metric trends, the $\alpha - \lambda$ metric results with varied BLS separation is illustrated in Figure 4.10. Again, the error bars indicate the standard deviation of the 150 simulations at each particular $\Delta p^{(L)}$ for the $\alpha - \lambda$ metric. Similar trends are observed where the GMJM/PF routine significantly outperforms the other predictors when subject to load transients of $\Delta p^{(L)} > 20$ W.

4.4 Experimental Validation

To demonstrate the efficacy of the integrated GMJM/PF prediction scheme, two separate unmanned ground vehicle studies were conducted. The unmanned ground vehicle was chosen for validation of the methodology as the loading profiles experienced during field operation remain stochastic and exhibit large transient jumps. In the first study, a single discharge test was performed with a Packbot UGV system operating in desert terrain. The results of this preliminary study were published in [17]. As the Packbot field study only contained one full discharge test, a laboratory stochastic terrain environment was constructed to physically simulate an unstructured environment and to allow for repeated/controlled UGV discharge tests. A modified National Instruments DaNI robot was used for these repeated in-house experiments. In both studies, no *a priori* knowledge of expected terrain was presumed, and run-time predictions only used online load characterization.

4.4.1 Packbot Case Study

For our first experimental study, a Packbot UGV was deployed and remotely controlled on generic desert terrain near Twentynine Palms, California. The Packbot used, shown in Figure 4.11, is a differentially driven tracked vehicle which also contains flipper arms for rough terrain navigation. To provide power during the experiments, the Packbot contained a 12 A-h UBI-2590 Li-ion pack which operated in parallel mode to provide a nominal pack voltage of 14.8 V. Hall-effect current sensors and voltage sensors recorded

the power loads of the 26 V main power bus at a rate of 1 kHz. Furthermore, the displacement of the Packbot was monitored via GPS at a rate of 1Hz. As a modified Packbot was deployed, the results of this study do not represent nominal Packbot performance.



Figure 4.11: Deployed Packbot maneuvering through diverse terrain during the discharge test.

To ensure significant transient loads were encountered during operation, the vehicle operator navigated the Packbot UGV over diverse terrain, shown in Figure 4.11. Generally, the terrain traverse could be classified as loose sand, organic foliage, gravel and larger stone obstacles. A predetermined course was selected such that the Packbot would traverse each terrain type to ensure diverse power loads. The vehicle operator remotely controlled the Packbot via a hand-held controller and maintained direct line-of-sight observation of the vehicle during the entire discharge test. Over the course of the discharge test, the Packbot ultimately traversed 762 meters of desert in 55.15 minutes, shown in Figure 4.12, on the single charge. A transient current spike of 19.65 A, resulting from attempting to breach particularly heavy foliage, ultimately resulted in battery shutdown.

priori characterization of this complex loading process is feasible.

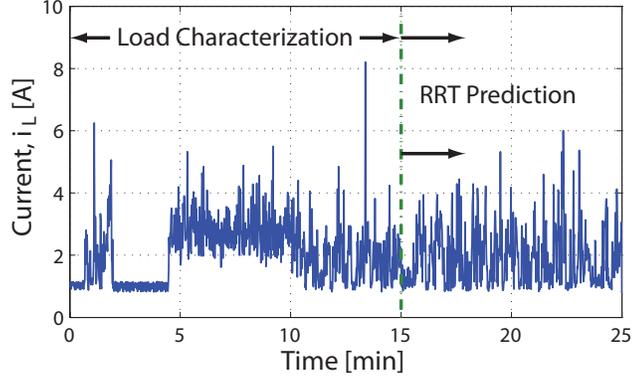


Figure 4.13: Measured Packbot bus current loads, during discharge study on unstructured terrain, used for load forecasting and run-time prediction.

To characterize the encountered load demands online, the self-supervised GMJM algorithm was implemented. To demonstrate the load scheme for the Packbot, the first 15 minutes of load data are used for self-supervised load characterization. As discussed in Section 3.3, the number of transient load clusters must be identified online due to the lack of *a priori* load information in complex mission environments. Iterative fitting of the Packbot current load GMM models and AIC evaluation identified eight transient load clusters, shown in Figure 4.14.

Upon the self-supervised identification of the number of load clusters, the respective GMM load model is selected to cluster load data. For characterized load of the Packbot case study at 15 minutes, the eight load cluster statistics are given in Table 4.1 and graphically illustrated in Figure 4.15(a). To physically motivate the self-supervised identified clusters, the GMJM al-

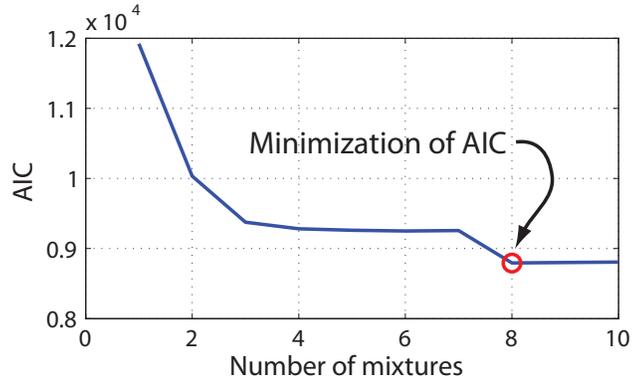


Figure 4.14: Unsupervised load cluster identification of Packbot loads for the first 15 minutes via AIC metric.

gorithm identified a low power cluster, cluster seven, with a mean of 23.66W. This particular cluster recognized the Packbot hotel loads which corresponds to a reported 25 W hotel load [17]. Furthermore, cluster seven has a mixture weight of 0.15 which corresponds to the actual time spent at idle, 15.6% of the first 15 minutes, by the Packbot.

Table 4.1: Self-supervised parameters for the GMM clusters of Packbot loads.

Mixture (k)	Mean (μ_k) [A]	Standard Deviation (σ_k) [A]	Mixture Weight (α_k)	Load Power [W]
1	2.67	0.47	0.20	69.42
2	3.39	0.92	0.09	88.14
3	1.14	0.12	0.13	29.64
4	1.53	0.24	0.15	39.78
5	1.13	0.02	0.08	29.38
6	2.07	0.29	0.18	53.82
7 - Hotel	0.91	0.07	0.15	23.66
8	5.54	1.56	0.01	144.04

After clustering the current data into respective load clusters, the load transition characteristics can be found via the maximum likelihood charac-

terization of the load Markov chain. To attempt to limit the computation requirements of the implemented particle filter routine, the prediction step size was set at one second. As such, the measured data, originally at a rate of 1 kHz, was resampled/clustered at 1 Hz. Given the 1 Hz cluster data, the Markov transition characteristics for the Packbot loads were found and graphically illustrated in Figure 4.15(b). Each grid square Figure 4.15(b) represents the probability of switching from a prior load cluster to a posterior load cluster given a time step of 1s.

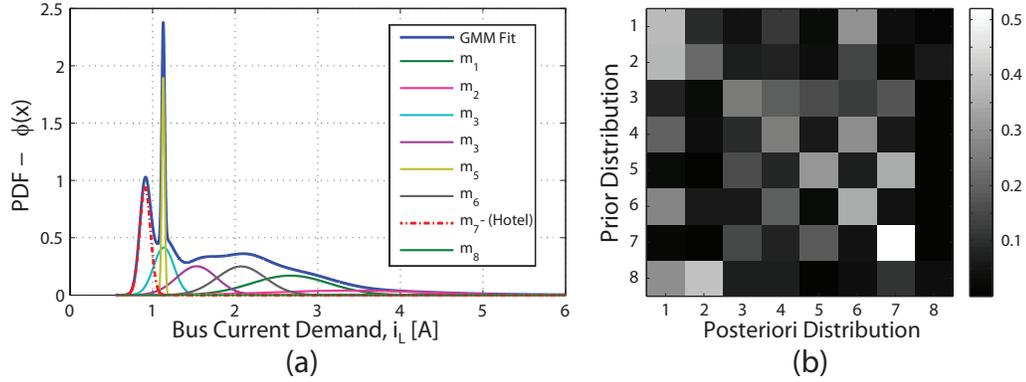


Figure 4.15: GMJM characterization of Packbot loads at 15 minutes. (a) GMM cluster components of current loads. (b) Markov transition matrix between load clusters.

With the characterized load presented above, the PF algorithm was employed for RRT prediction for the Packbot system at 15 minutes. For prediction fidelity comparison, the EKF predictor was also utilized for RRT prediction. Using the shutdown conditions discussed in [17], the Packbot battery terminal voltage was forecast until shutdown conditions were reached for each particle, seen in Figure 4.16. For each prediction step, the probability of

system shutdown was evaluated via the reliability integrate discussed in Section 4.1.1. Both the RRT PDF and CDF, predicted at 15 minutes into the Packbot mission, are presented in Figure 4.17.

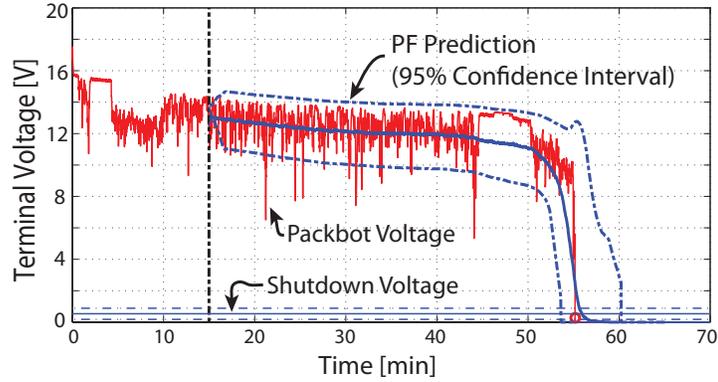


Figure 4.16: Predicted Packbot battery voltage via the GMJM/PF scheme at 15 minutes mission time with 95% confidence intervals.

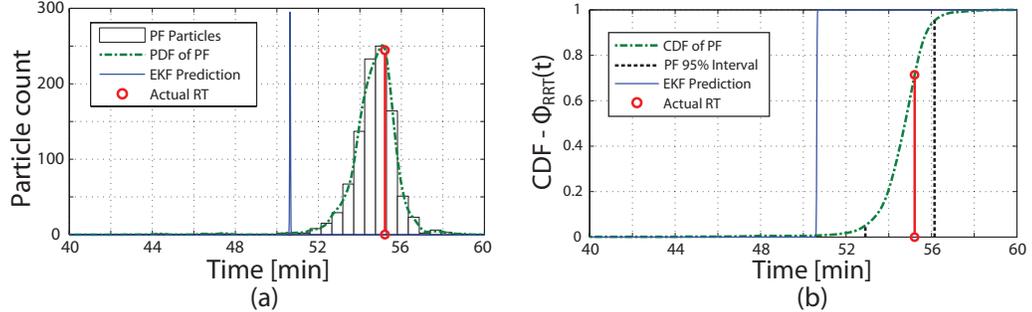


Figure 4.17: Statistical run-time predictions made at 15 minutes into mission as compared to the EKF. (a) PDF prediction. (b) CDF prediction.

To evaluate the prediction fidelity of the PF over the entire Packbot mission profile, the prognostic horizon methodology introduced in Section 2.1.4 was utilized. Remaining run-time predictions were conducted at five minute intervals over the course of the Packbot mission ranging from 10 minutes to

50 minutes. Prior to each prediction, the statistical characteristics of the measured loads were self-characterized via the GMJM process. The PF run-time predictions over the entire prognostic horizon were compared to the ground truth Packbot run-time of 55.15 minutes. The prognostic horizon predictions for both the EKF and the PF are shown in Figure 4.18 with additional 95% confidence error bounds for the PF predictor. The EKF confidence intervals were not included for clarity, but remained under one minute for the entire prediction horizon.

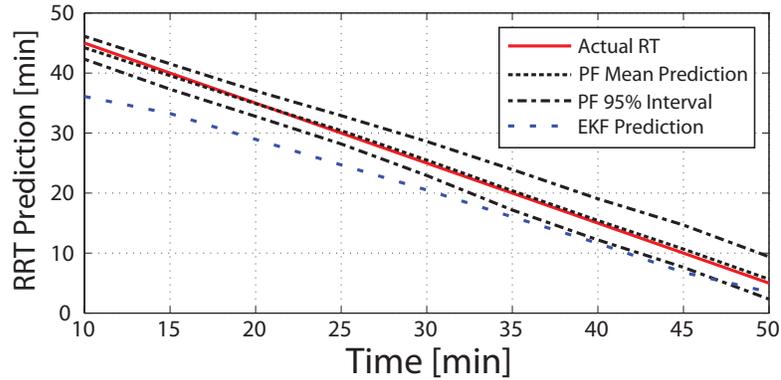


Figure 4.18: Prediction fidelity over the prognostics horizon for the EKF predictor and the PF/GMJM prediction scheme.

As is clearly illustrated in Figure 4.18, the EKF predictor returns conservative run-time predictions, as discussed in Section 2.1, and does not approach the ground truth run-time value until the final 10 minutes of vehicle operation. Contrastingly, the PF/GMJM prediction algorithm 95% confidence interval bounds the ground truth value over the entire prognostic horizon. Additionally, the CRA for the Packbot implementation of the PF was 0.8767 over

the interval of $[0,1]$ with unity implying perfect prediction. Of course, prediction accuracy results as a trade-off from computational requirements. The PF implementation over the prediction horizon required 30.25 ± 10.41 s for each prediction whereas the EKF only required 3.65 ± 0.23 s on a 2.6 GHz processor. Resultantly, the choice of prediction routine remains application dependent and should be influenced by the expected load transient characteristics, as demonstrated in Section 4.3.

4.4.2 In-house Small Ground Vehicle Test Stand Study

To provide statistical significance of the integrated GMJM/PF algorithm predictions, an in-house stochastic terrain simulator was constructed for traversal by a small ground robot. The terrain simulator allows for repeated discharge tests with similar power load statistics/transients, such that multiple realizations of the UGV discharge stochastic process are generated. A small ground robot was developed with wall following capabilities for repeated traversal of the terrain environment. The following section introduces the UGV system and the terrain setup with applications to run-time prediction.

4.4.2.1 DaNI Unmanned Ground Vehicle

For the repeated indoor studies, a second generation National Instruments DaNI robotic ground vehicle platform was used, shown in Figure 4.19 with significant upgrades. The small UGV, weighing 3.6 kg, is differentially driven via two geared Pitsco W39083 DC motors with peak torque of 2.12

N-m each, with an additional rear omni-directional passive wheel for vehicle stability. Additionally, an onboard sbRIO-9632 with a coupled 400 MHz processor and Xilinx Spartan FPGA provides for online processing/control and data acquisition through 6 differential analog voltage channels, respectively. To provide power for untethered operation, the DaNI UGV is equipped with a ten cell 12V 3000 mA-h NiMH battery pack which yields a reported operation time of one hour with motorized locomotion. The deployed NiMH battery packs were experimentally parameterized in studies presented in Appendix A.1.

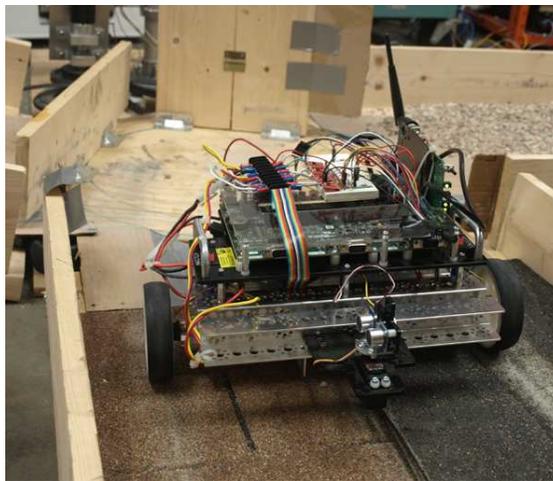


Figure 4.19: Small unmanned differentially driven ground vehicle, also known as the NI DaNI UGV, used for laboratory discharge studies.

Vehicle upgrades include an additional sensor package for online power measurements, remote distance sensors for position control, and a wireless communications system, see Figure 4.19. For full untethered operation, the DaNI robot system was upgraded to include onboard 802.11g wireless com-

munication. This was achieved via an attached Linksys WRT54GL router used for remote data logging and inter-device communication. A wireless network bridge was formed with the laboratory local network to allow for online recording of all UGV measured data at a rate of 10 Hz. Furthermore, wireless communication between the supervisory computer, the UGV control board, and the test stand controller provided for synchronous operation of all systems. To monitor online power demands of the UGV, four Hall-effect current sensors were installed for battery, motors and bus/peripheral current monitoring. Additionally, battery terminal voltage and bus voltage measurements were logged directly via operational amplifier voltage dividers. To ensure proper path following, two Parallax ultrasonic distance sensors were installed laterally left on the front and rear of the vehicle. These distance sensors allowed for controlled wall following. Moreover, to determine the end of the terrain assembly, two infrared distance sensors were installed facing forward/reverse.

4.4.2.2 Automated Terrain Environment for UGV Testing

The goal of the automated terrain environment is to mimic field terrain in a laboratory setting for controlled UGV discharge studies. As the size of the indoor laboratory space was limited, the terrain environment was designed for both modularity of terrains and automated terrain selection. Figure 4.20(a) schematically illustrates the terrain environment with the three terrain modules and a turntable assembly to change terrain. To replicate the variety of terramechanical properties encountered during the desert terrain Packbot

study [17], three distinct terrain types were selected: loose gravel, rough shingle bumps, and a smooth inclination. Loose gravel serves as a deformable terrain that readily induces vehicle slip, whereas the rough bumps and the inclined terrain each increase the rolling resistance and grade resistance of the UGV, respectively [75]. The constructed terrain environment is illustrated in Figure 4.20(b).

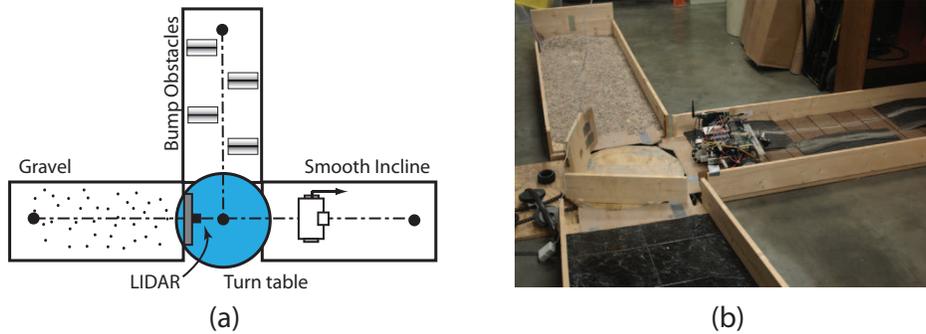


Figure 4.20: (a) Schematic of the automated terrain environment assembly for UGV run-time testing. (b) Implementation of the terrain environment.

As encountered in the Packbot field study and other experiments [82], diverse intervals of uniform terrain exist in unstructured environments. To experimentally replicate varied terrain intervals, the turntable assembly is randomly driven by a Markov chain. The transition matrix for the Markov chain was selected to match the rate of change of load clusters measured during the Packbot field study [17]. Upon return of the UGV to the turntable, a realization of the Markov chain, contingent on the previous terrain, is generated to determine the subsequent terrain. The control of position of the turntable is achieved via an integrated rotary encoder/DC motor assembly and a NI cRIO

controller. The cRIO generates the realizations of the Markov chain.

The automated run-time experiment process requires the UGV to longitudinally traverse varied terrain until battery shutdown. With a charged battery pack, the UGV travels longitudinally forward until the forward IR sensor is triggered, after which the direction of the vehicle reverses. Upon triggering the rear IR sensor, the turntable Markov chain is realized and a new terrain selected. Finally, to track vehicle displacement and provide for velocity estimates, an URG-04LX LIDAR system was installed on the terrain assembly to record vehicle position at a rate of 10 Hz. The LIDAR system remains fixed to the back of the turntable assembly, such that the displacement of the vehicle on each terrain could be recorded. The average displacement data from the center -2.5 to 2.5 degree LIDAR cone angle was used for the vehicle displacement value for each step in time.

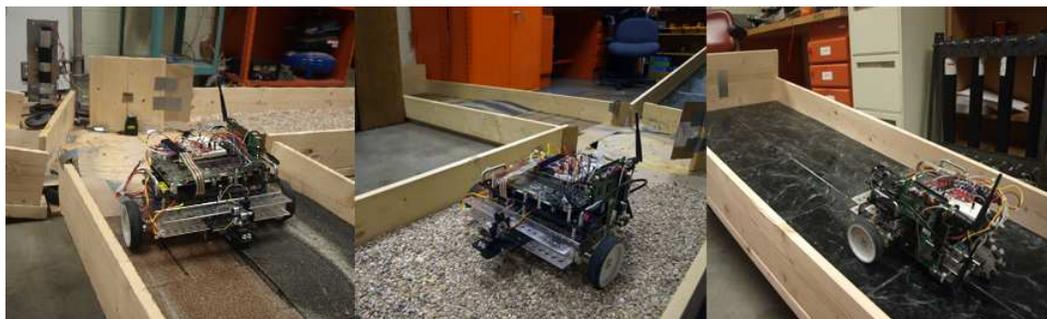


Figure 4.21: Close-up of the small UGV on each terrain type: (a) Rough shingles. (b) Loose gravel. (c) Smooth incline.

4.4.2.3 Run-time Prediction Analysis

For analysis of the GMJM/PF prediction scheme, twenty five discharge tests were performed with the UGV on the automated terrain environment. While the used NiMH packs permit for 3000 mA-h of charge, UGV discharge tests were conducted with initial battery charges of 1500 mA-h to reduce experimentation time from approximately 2.5 hours to 1.25 hours. Furthermore, as seen in Figure 4.22, the mean, variance, skewness, and kurtosis sample statistics for run-time converge before the twenty fifth experiments. Due to repair costs and time constraints, convergence of these statistical moments was deemed sufficient for statistical analysis [143]. In an attempt to mitigate variability between test iterations, vehicle maintenance was performed between discharge tests to minimized uncertainty associated with hardware deterioration. Finally, the principal contribution of uncertainty in these discharge studies was the locomotion power demands as peripheral electronic power draw required only 23.66 ± 0.84 W.

The results of each UGV run-time test are illustrated in Appendix B. Provided is tabulated information on the final run-time of each test and the final measured battery voltage which is deemed to be the shutdown voltage. Additionally, the peak measured current for each test was noted to further illustrate the differences in transient load demands between tests while operating on identical terrain. Furthermore, Table B.3 illustrates the breakdown of the fraction of time spent on each terrain type.

As discussed, the individual path realizations, during each discharge

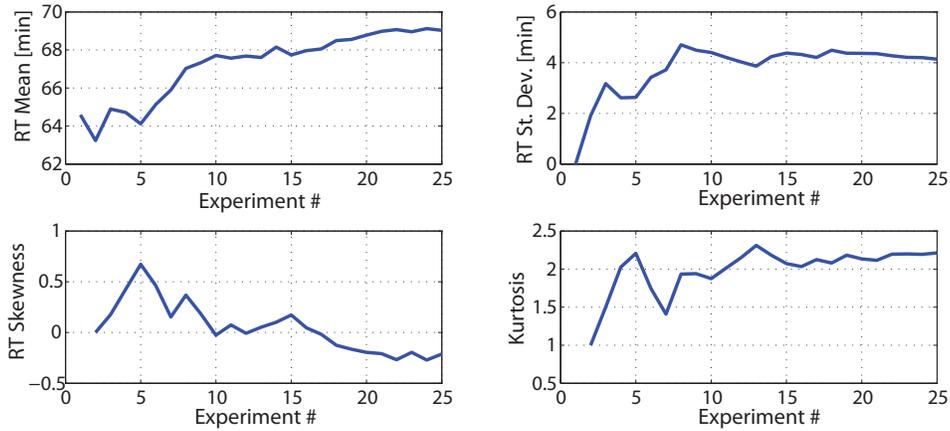


Figure 4.22: Convergence of the mean, variance, skewness and kurtosis imply sufficient number of experimental UGV discharge studies.

test, differ due to the *a priori* specified terrain Markov process which results in unique transient power loads. Contrastingly, the steady state characteristics of the measured power loads demonstrate similarity between each UGV discharge test. Figure 4.23(a) illuminates the transient power load differences of the first 15 minutes between experimental studies, whereas Figure 4.23(b) demonstrates the measured battery power demands of the first three discharge studies as normalized histograms. As a result, each of the experimental studies serves as a realization of the UGV discharge stochastic process. Consequentially, to statistically demonstrate run-time prediction fidelity, the GMJM/PF scheme should successfully predicted run-time for each transient realization of the test stand.

The run-time results from each discharge experiment are shown as a histogram in Figure 4.24 with a minimum UGV run-time of 61.72 minutes and maximum run-time of 76.03 minutes. To demonstrate individual pre-

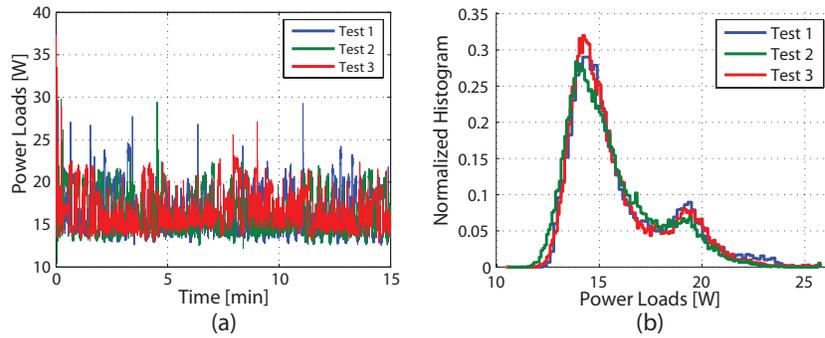


Figure 4.23: (a) First fifteen minutes of battery power demands for three UGV discharge studies (b) Steady state battery power loads displayed via a normalized histogram.

diction capabilities for the DaNI UGV study, a PF battery voltage forecast and predicted RRT distribution are illustrated in Figure 4.25. To analyze the accuracy of the prediction methodologies, a prognostic horizon analysis was performed for each UGV run-time test from 10 minutes to 55 minutes with the modified Peukert’s predictor discussed in Section 2.1, the EKF predictor, SOC linear regression forecasting, and the proposed GMJM/PF prediction scheme. The prognostic horizon for the first UGV run-time experiment is shown in Figure 4.26.

To evaluate the prediction fidelity over the prognostic horizons of each UGV run-time test, both the cumulative relative accuracy and cumulative $\alpha-\lambda$ metrics were evaluated for each test. The CRA test, introduced in Section 2.1.4, assesses the total difference of mean run-time predictions as compared to the ground truth runtimes and remains bounded on the interval, $[0,1]$. Predictions with CRA scores near unity consistently predict near the ground

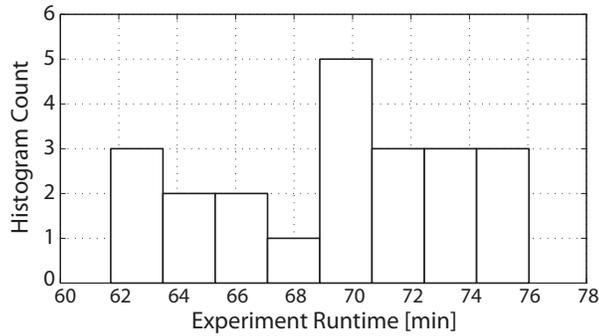


Figure 4.24: Experimental runtimes for twenty-five UGV discharge studies.

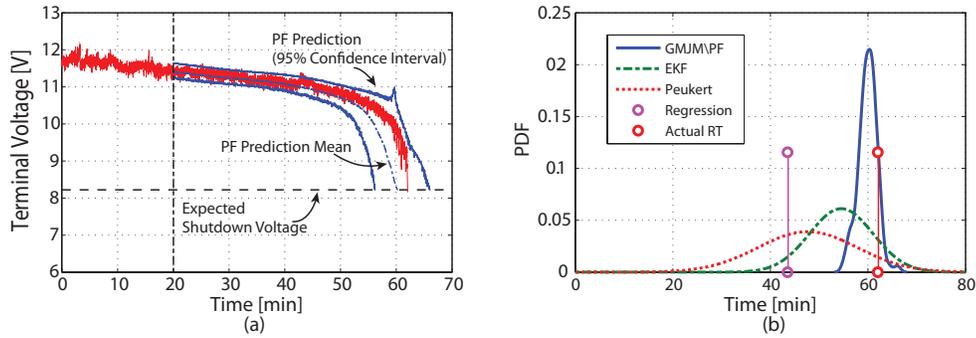


Figure 4.25: Example GMJM/PF prediction of RRT at 20 minutes into experiment one. (a) PF forecast of battery voltage as compared to measured voltage. (b) PF predicted run-time PDF.

truth run-time. Contrastingly, the $\alpha - \lambda$ metric assesses the fidelity of the predicted run-time PDF as compared to a shrinking confidence interval. Again, the $\alpha - \lambda$ metric is bounded on the $[0,1]$ interval with unity implying perfect prediction over the entire prognostic horizon. The prognostic parameters for $\alpha - \lambda$ were selected as $\beta = 0.35$ and $\lambda = 0.1$ corresponding with [66]. The cumulative statistics of the CRA and $\alpha - \lambda$ metrics for all discharge experiments are presented in Figure 4.27.

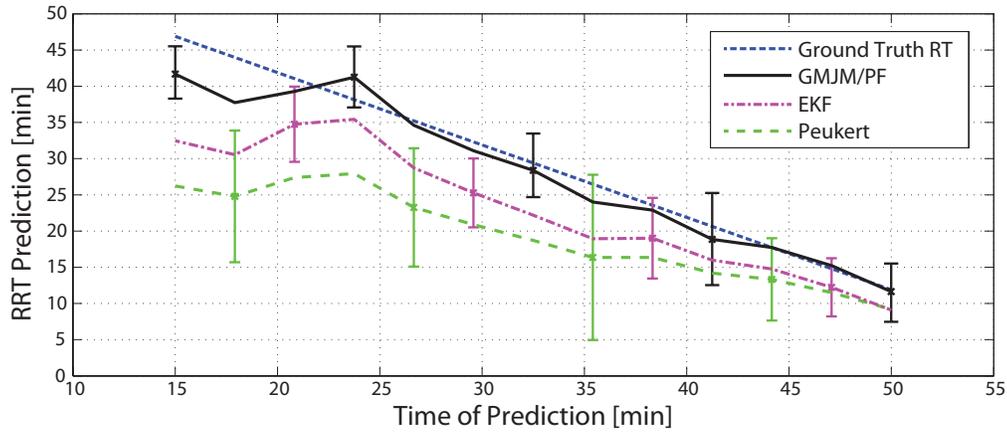


Figure 4.26: Prognostics horizon for test one comparison of prediction methodologies for the in-house UGV studies.

Clearly illustrated in Figure 4.27(a) and (b) is lack of prediction fidelity acquired with SOC regression forecasting. As the encountered power demands exhibit significant transients, the SOC estimates do not exhibit local linear trends and run-time predictions thusly remain inaccurate via both prediction metrics. While the modified Peukert’s law/current counting methodology prediction scheme outperforms regression, uncertainty in prediction fidelity is high. Furthermore, inspection of the prognostic horizon for each experimental discharge study indicates conservative prediction. Likewise, the EKF predictor exhibits conservative prediction biases over the entire prognostic horizon, as seen in Figure 4.27.

The GMJM/PF prediction scheme outperforms Peukert’s and regression by significant margins and demonstrates a statistically significant improvement over the EKF predictor for both the CRA and $\alpha - \lambda$ metrics. A two-sample t-test was performed to investigate if the CRA improvement of the

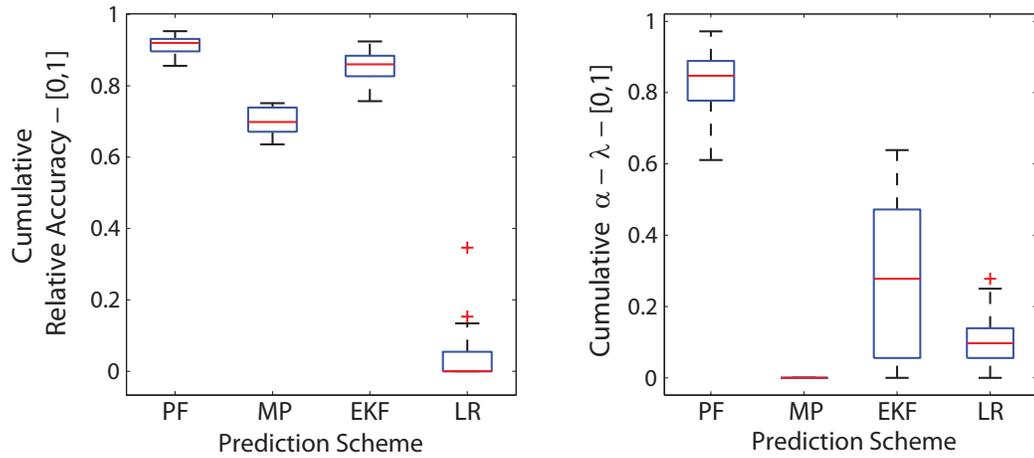


Figure 4.27: Prediction fidelity over all UGV run-time tests. (a) Cumulative relative accuracy (b) Cumulative $\alpha - \lambda$ metric.

GMJM/PF predictor over the EKF was statistically significant. The p-value of the null hypothesis, where the GMJM/PF and EKF predictors exhibit identical prediction characteristics, was found to be 6.36×10^{-6} well below the 5% significance level. As such, the null hypothesis is rejected and the GMJM/PF CRA improvement is statistically significant.

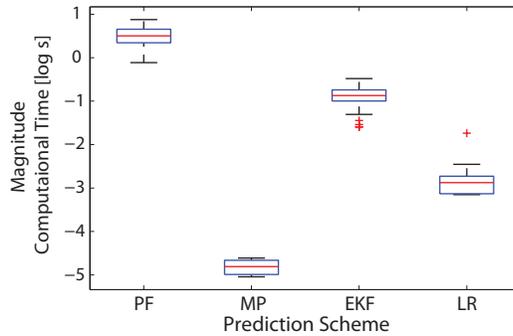


Figure 4.28: Comparison of algorithm computation times for prediction.

Gains in prediction fidelity of the GMJM/PF prediction scheme are

not without cost, however. As illustrated in Figure 4.28, the particle filter requires 3.39 seconds (or 2264%) more mean computation time than the EKF prediction methodology as computed on a 2.6 GHz Intel Core i7 processor. The significant increase in required onboard computational power potentially invalidates the gains in prediction accuracy for small scale applications. However, as the results from the Monte Carlo study in Section 4.3 indicate, the power transient jump range encountered in the DaNI UGV studies, a study average of 5.98 W, are near the lower bound where the GMJM/PF benefits become particularly apparent. The experimental results corroborate with the lower bound found in the Monte Carlo studies. Furthermore, as was demonstrate in the Packbot experimental case study from Section 4.4.1, the EKF prediction significantly underperformed the GMJM/PF algorithm due to the power cluster jumps of 120.38 W.

Chapter 5

Online Mission Reliability Assessment

Quite often, mobile systems must achieve mission objectives using finite onboard energy. For instance, electric vehicles transport a driver from a starting location to desired final location, and must do so without exhausting onboard energy [149]. Unmanned ground vehicles, whether teleoperated or strictly autonomous, are deployed with *a priori* specified mission objectives, such as reconnaissance/mapping of a building [12] or extraplanetary sample acquisition [150]. Again, the UGVs must accomplish these objective within the constraints of the onboard energy. With the GMJM/PF prediction scheme, a mobile system can predict the battery remaining run-time. However, the predicted run-time PDF does not independently garner insight into the probability of successfully completing all mission objectives within the constrained run-time.

As such, in this chapter, an online mission reliability assessment scheme is presented which evaluates the probability of mission completion (PoMC) based on mission objectives and energy storage. The multivariate reliability theory methods discussed in Section 2.4 are directly applied here for computing mission reliability. A mobile system can be viewed as a stochastic

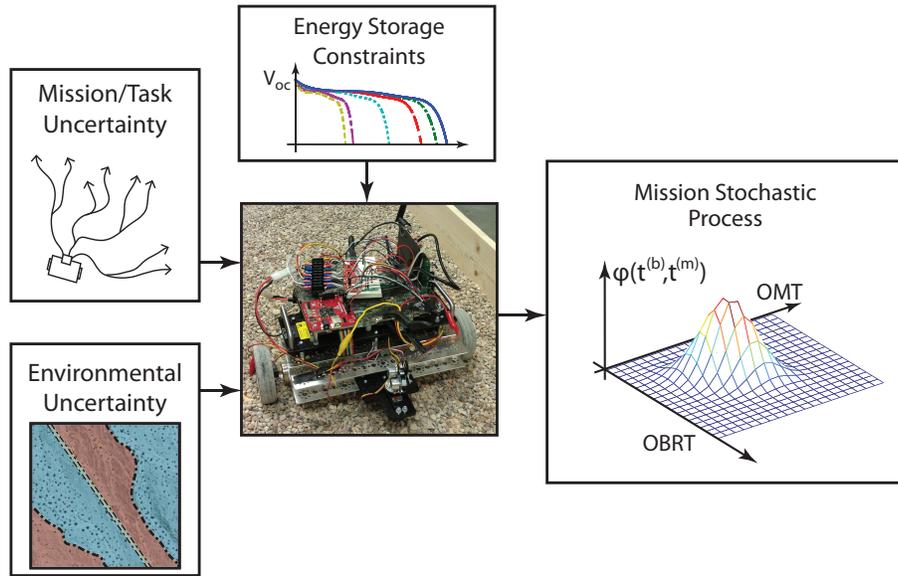


Figure 5.1: Conceptualized mobile robotic system as a general stochastic process.

process with uncertain input power loads and mission demands, all of which are constrained by available onboard energy storage. A generic mobile system mission process is depicted in Figure 5.1. The mission process is comprised of two dependent random processes, the overall mission time (OMT) and the overall battery run-time (OBRT). The mission time random process is comprised of the summation of the overall time for tasks (OTT) and the overall drive time (ODT). The OBRT results from the stochastic discharge process detailed in Section 4.1.2. The OMT and OBRT remain correlated through the mission process. For example, an arduous mission profile with high rates of travel results in a decrease in the OMT, but also a decrease in OBRT due to the higher currents necessary for travel [21]. Given a characterization of the

mission process depicted in Figure 5.1, the PoMC can be estimated through the probability of OMT exceeding OBRT.

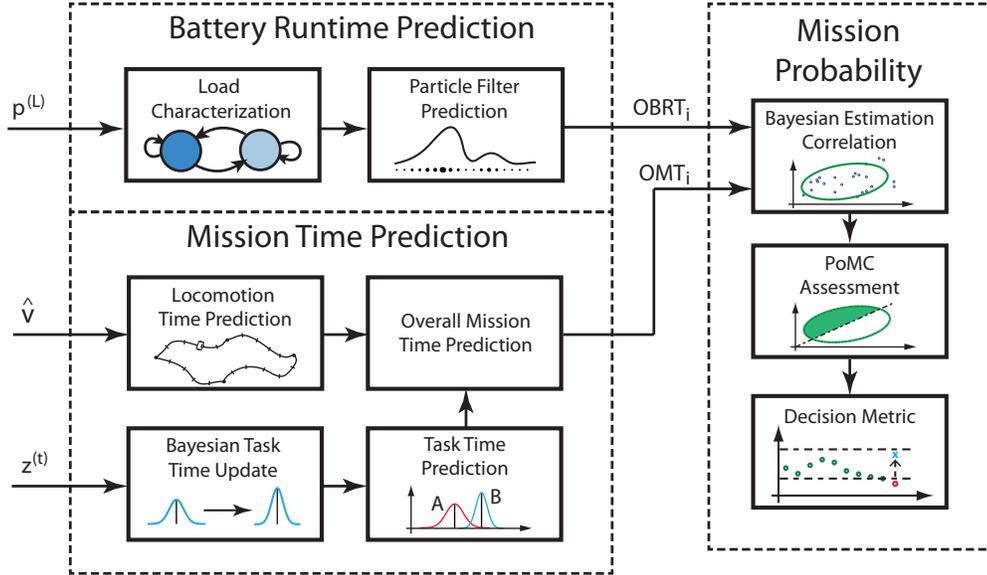


Figure 5.2: Integrated mission reliability assessment/decision algorithm.

The proposed algorithm, illustrated in Figure 5.2, adopts a piecewise approach with two independent predictors of the marginal OMT and OBRT processes. Prediction of the marginal OMT process statistics $(\hat{\mu}_{MT}, \hat{\sigma}_{MT}^2)$, which is comprised of the summation of OTT statistics $(\hat{\mu}_t, \hat{\sigma}_t^2)$ and ODT statistics $(\hat{\mu}_d, \hat{\sigma}_d^2)$, is discussed in Sections 5.1.1 and 5.1.2, respectively. Prediction of OBRT was discussed at length in Chapter 4. However, for bivariate distribution characterization, the correlation between each marginal distribution must be known [143]. Using successive prediction pairs $(\overline{OBRT}_i, \overline{OMT}_i)$, however, the mission correlation estimate can be improved via recursive Bayesian updating. Bayesian correlation updating is discussed in Section 5.2.3.

With marginal predictions of OMT and OBRT and the estimate for process correlation, an estimate for the bivariate mission process can be formulated. Representation of the bivariate process is discussed in Section 5.2.2. Finally, the PoMC can be estimated, as discussed in Section 5.2.4. As the bivariate characterization only involves OMT and OBRT, the PoMC estimate presented in this work discounts terrain traversibility and component failure considerations. While component failure and vehicle mobility result in system failure [151], all mobile system ultimately fail due to energy exhaustion. Resultantly, only energy concerns are addressed in this work with extensions to vehicle mobility and component failure anticipated as future work. Finally, the experimental studies, discussed in Section 4.4, are used to demonstrate the efficacy of the proposed methodology.

5.1 Evaluation of Overall Mission Time

Generally, a mobile system mission profile consists of time for travel and distinct tasks. Resultantly, mission times could range from several minutes, in the case of mini-unmanned aircraft system performing room mapping [152], to days for the Hyperion robot exploration of the Atacama desert [106]. While travel trajectories and necessary tasks which compose the overall mission can be specified prior to system deployment, uncertainty associated with the environment could mandate a real-time updates of the planned path or essential tasks. For example, robotic search and rescue task time depend on the environment to search [153]. As such, the OMT prediction methodology must

consider variable task time statistics and uncertainty associated locomotion velocities.

5.1.1 Task Time Estimation

In this dissertation, all goals, with the exception of vehicle locomotion, are considered mission tasks. Some tasks performed during a mission, such as obstacle avoidance or visual surveillance, reoccur at variable intervals. Contrastingly, some missions entail a single task, such as sample acquisition or a rescue operation [12]. Individual task time statistics are assumed to follow a normal distribution for computational efficiency and to exhibit statistical independence [21]. Provided Gaussian task times and assuming that each task is statistically independent, the overall mean mission task time estimate is found via the summation of estimated task means, as follows,

$$\hat{\mu}_t = \sum_{i=1}^n \omega_i \hat{\mu}_{t_i} \quad (5.1)$$

where n is the number of distinct tasks, and ω_i and $\hat{\mu}_{t_i}$ are the number of repetitions and the current mean estimates for the i 'th task, respectively. Similarly, the task time variance estimate is found accordingly,

$$\hat{\sigma}_t^2 = \sum_{i=1}^n \omega_i \hat{\sigma}_{t_i}^2 \quad (5.2)$$

where $\hat{\sigma}_{t_i}^2$ is the current variance estimates for the i 'th task.

Task characterization prior to executing a mission provides an *a priori* estimate of required time for each necessary mission objective, shown in Figure 5.3. Nevertheless, *a priori* characterization in a laboratory setting does

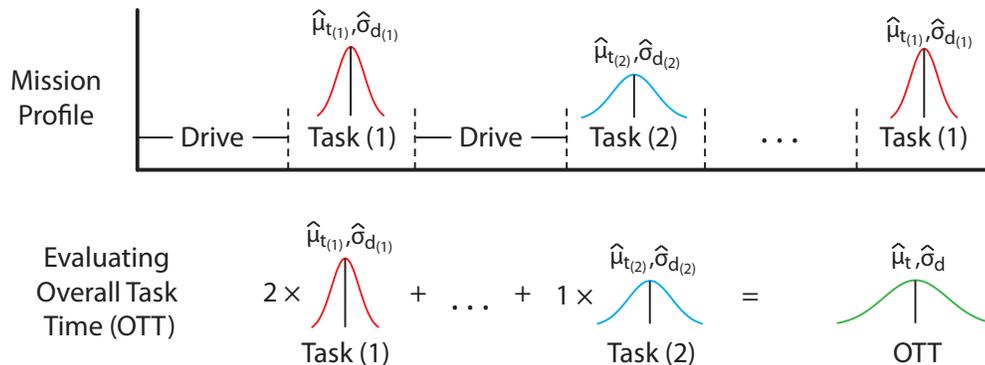


Figure 5.3: Analytically computation of OTT via summation of normal random variables.

not necessarily accurately characterize field task time statistics. Contrastingly, these *a priori* measurements can serve as an estimate for the initial task time distribution (an informative prior assumption) for sequential Bayesian parameter updating given additional online measurements of task times in the field. Online updating of task times is reserved for future work, but the update form is expected to take the form a recursive Bayesian update structure for a unknown mean and variance parameters (a normal-inverse-gamma conjugate prior distribution) [64]. Finally, particular tasks, such as obstacle traversal by a UGV, may exhibit non-Gaussian statistics due to repeated trials required to overcome an impediment [26]. However, the cumulative summation of all independent mission tasks tends towards a normally distributed process, due to the Lyapunov condition of the central limit theorem [143].

5.1.2 Overall Travel Time Prediction

As introduced in Section 2.4, prediction of travel time remains an open problem, in particular for automotive applications [154]. Existing methodologies utilize data-driven techniques, such as least squares regression, support vector regression, ARIMA prediction, and neural networks to predict travel time via historical data [155, 156] and measured travel velocities [157]. As utilized in [21], an exponentially weighted moving average (EWMA) scheme, also known as an ARMA(1,1) model, was adopted for velocity forecasting. While other methodologies provide potential gains in forecasting fidelity and can be investigated in future works, the EWMA was selected for computational efficiency.

The EWMA model weights velocity data such that the recently observed/estimated velocities contribute significantly to the sample mean and variance estimates [158]. Weights on past samples decay via an exponential forgetting factor, λ_v [76]. Additionally, the mean estimate can be computed recursively,

$$\hat{\mu}_v = \lambda_v v(k) + (1 - \lambda_v) \hat{\mu}_{v_{k-1}} \quad (5.3)$$

where $\hat{\mu}_{v_{k-1}}$ is the previous EWMA mean estimate and $v(k)$ is the latest velocity measurement/estimate. Similarly, the EWMA variance estimate can be computed,

$$\hat{\sigma}_v^2 = (1 - \lambda_v) \left[\hat{\sigma}_{v_{k-1}}^2 + \lambda_v (v(k) - \hat{\mu}_{v_{k-1}})^2 \right] \quad (5.4)$$

where $\hat{\sigma}_{v_{k-1}}^2$ is the previous EWMA variance estimate. The forgetting factor was selected to correspond with [21] such that $\lambda_v = 0.002$.

With the EWMA velocity statistics $(\hat{\mu}_v, \hat{\sigma}_v^2)$, predicting the mean distance of travel for a specified time step is found via the summation over time of the independent velocity random variables [156]. Provided a deterministic desired mission distance, D_m , the remaining drive time statistics can be found via the intersection of the predicted cumulative distance distribution with the mission distance, as shown in Figure 5.5. Similar analysis can be conducted provided uncertainty in mission distance without the benefits of analytical tractability.

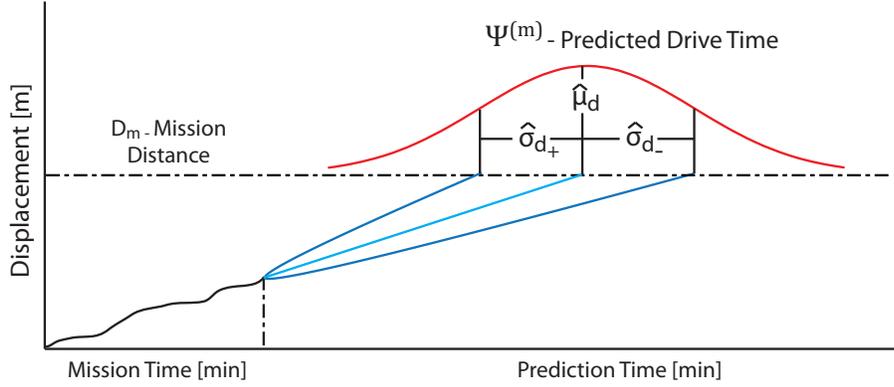


Figure 5.4: Analytically determining remaining drive time via cumulative velocity forecasting.

As seen in Figure 5.5, the mean cumulative velocity forecast intersects the mission distance via a linear relationship, as follows,

$$\hat{\mu}_d = \frac{D_m}{\hat{\mu}_v}, \quad (5.5)$$

where D_m is the current remaining mission distance and $\hat{\mu}_d$ is the predicted mean drive time. To derive the analytical expression for mission time variance, the intersection of standard deviations of the forecasted cumulative velocity

statistics with the mission distance can be found, as depicted in Figure 5.5.

Mission time variance is expressed analytically as,

$$\hat{\sigma}_d^2 = \frac{\hat{\sigma}_v^2}{4\hat{\mu}_v^4} \left[\hat{\sigma}_v + \sqrt{4D_m\hat{\mu}_v + \hat{\sigma}_v^2} \right]^2. \quad (5.6)$$

Due to the nonlinear transformation associated with variance prediction, the predicted drive time distribution exhibits non-zero skewness. To analyze the significance of the non-Gaussianity of the drive time distribution, Monte Carlo simulations were conducted of the stochastic travel process depicted in Figure 5.5. The simulations were performed assuming a rate of travel of 0.25 m/s with a velocity standard deviation of 0.1 m/s with a goal distance of 100 m. As seen in the normal probability plot in Figure 5.5, only the 1% tails of the distribution are not approximated via the normal. Resultantly, the analytically predicted drive time distribution is assumed to follow a normal distribution.

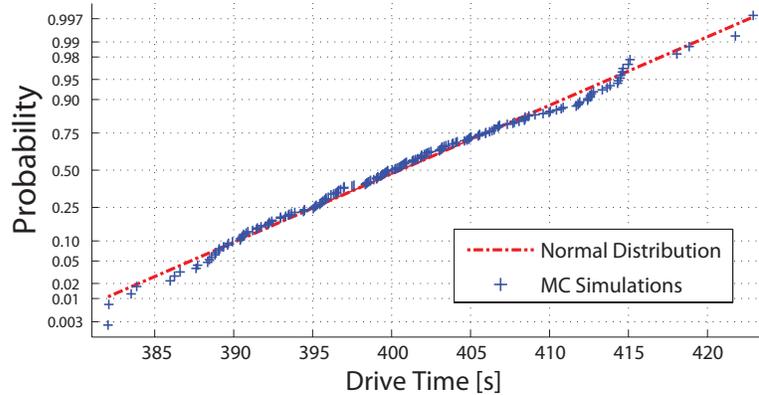


Figure 5.5: Monte Carlo simulation demonstration of normality of analytically predicted drive-time distribution.

5.1.3 Characterization of Overall Mission Time

With predictions for task times $(\hat{\mu}_t, \hat{\sigma}_t^2)$ and drive times $(\hat{\mu}_d, \hat{\sigma}_d^2)$, the OMT statistics can be computed. Resulting from the assumed Gaussianity of both task and drive statistics, the summation of the task means and drive time means from equations (5.1) and (5.5), respectively,

$$\hat{\mu}_{MT} = t^{(c)} + \hat{\mu}_d + \sum_{i=1}^n \hat{\mu}_{t_i}, \quad (5.7)$$

yields an estimate for the mean OMT, $\hat{\mu}_{OMT}$, with $t^{(c)}$ as the current mission run-time of the mobile system. Furthermore, the OMT variance estimate is computed via direct summation of task and drive time variances given by equations (5.2) and (5.6), respectively. The OMT variance estimate is given by,

$$\hat{\sigma}_{MT}^2 = \hat{\sigma}_d^2 + \sum_{i=1}^n \hat{\sigma}_{t_i}^2. \quad (5.8)$$

5.2 Mission Completion Assessment

The preceding discussion has demonstrated methods for online prediction of OMT and OBRT individually. Traditional univariate reliability methods, predicting the probability of OMT exceeding OBRT, disregard the inherent correlation between the OMT and OBRT processes [21]. As such, a multivariate statistical representation of a combined OMT/OBRT process must be adopted for reliability mission assessment. The following sections conceptually introduce the bivariate mission process and demonstrate the Bayesian correlation estimation method.

5.2.1 Determining a Mission Process Statistical Model

Due to the correlation between the OMT and OBRT random processes, the overall mission random process is represented via a bivariate distribution. Even if each marginal process displays normally distributed statistics, no guarantee exists to ensure the normality of the bivariate mission distribution [143]. To determine the expected structure of the multivariate mission process, Monte Carlo discharge simulations were conducted for an UGV operating in a stochastic environment using the model introduced in Section 6.2.1.

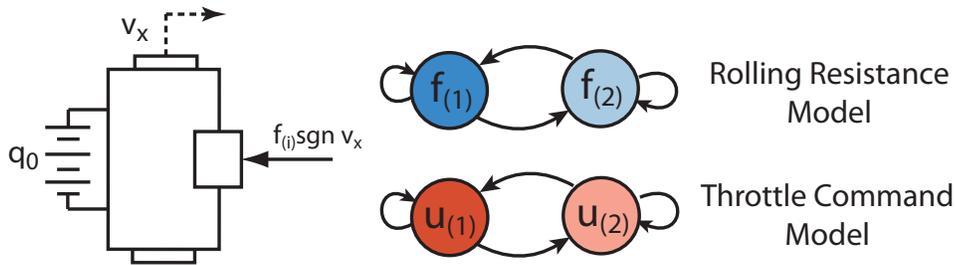


Figure 5.6: Monte Carlo simulation of simplified stochastic throttle commands and terrain properties with deterministic battery charge.

To minimize study degrees of freedom, the simulation robot was issued only two throttle commands (10% and 90%) as a random process defined by a second-order Markov chain. The second-order Markov chain characteristics are given in Section 4.3.1. As a result, the steady state probabilities for throttle commands were directly specified as $pi_1^{(t)} = 0.3$ and $pi_2^{(t)} = 0.7$, with a transition rate of $\lambda_t = 0.99$. Furthermore, the UGV traverses two terrains with negligible slipping that have rolling resistance coefficients of 0.01 and 0.3. Again, the terrain characteristics are defined by a second-order Markov pro-

cess with a transition rate of $\lambda_r = 0.99$ and $\pi_1^{(r)} = 0.5$ and $\pi_2^{(r)} = 0.5$ steady state probabilities, respectively. Each simulation UGV begins with 1500 mA-h of charge.

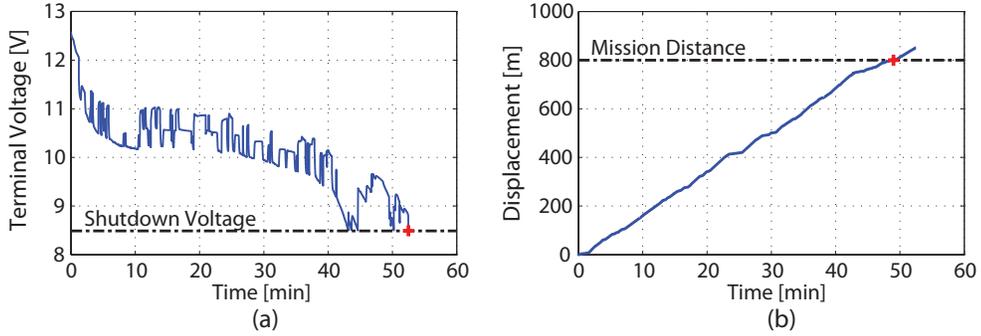


Figure 5.7: Single mission realization from the Monte Carlo simulation study. (a) UGV Battery Voltage. (b) UGV Cumulative Distance.

Monte Carlo analysis consisted of longitudinal driving at the indicated stochastic throttle rates over the two defined terrains until transient battery shutdown, shown in Figure 5.7. Shutdown occurs either due to state-of-charge exhaustion or the surpassing of the shutdown voltage by the battery terminal voltage [17]. Additionally, assuming a desired mission distance objective of 800 meters, each Monte Carlo realization provides the time necessary to reach the arbitrary goal. For 500 simulation realizations, the OMT and OBRT results are shown in Figure 5.8. Also depicted in Figure 5.8 to illustrate the importance of correlation estimation are the results from the DaNI UGV studies. The displayed OMT values indicated the time required for the DaNI UGV to travel 650 meters.

The Mardia normality test was employed to determine if a bivariate

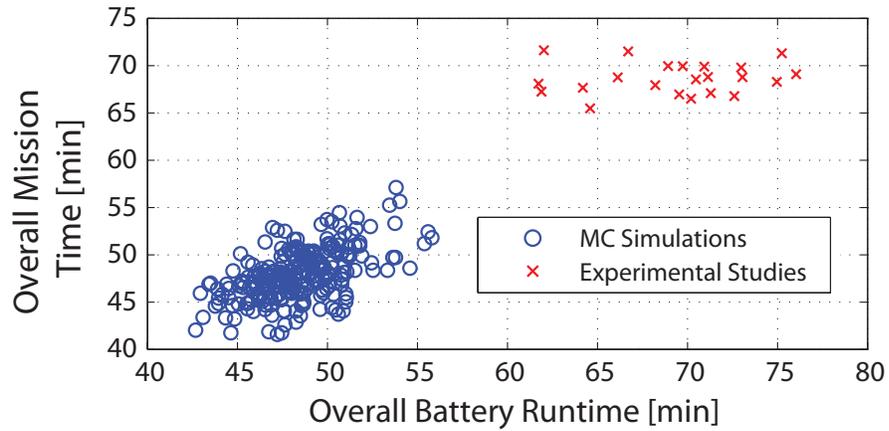


Figure 5.8: Results of a Monte Carlo simulation study of a UGV operating in a stochastic environment.

normal approximation suites the mission random process [143]. Akin to a univariate normality tests, the skewness and kurtosis of the multivariate data provides for a measure of similarity to a normal distribution. Assuming a null hypothesis of bivariate normality, the skewness and kurtosis statistics must not demonstrate significant divergence from their parameter distributions of the chi-squared and standard normal distributions, respectively. Provided p-values for each statistic do not demonstrate significance, the null-hypothesis is not rejected and the mission process is assumed to have a bivariate normal distribution. For the skewness statistics, the 5% significance level for the chi-squared distribution is 7.78. This value is not exceeded by the Mardia statistic for skewness of 5.51. As such, the null hypothesis remains unrejected due to the multivariate skewness. Likewise for the kurtosis statistic, the 5% critical value for the standard normal is 1.28, that again is not exceeded by the magnitude of Mardia statistic for kurtosis of 0.23. Resultantly, multivariate skewness and

kurtosis indicate the mission process can be appropriately represented as a bivariate normal distribution.

5.2.2 Bivariate Representation of Mission Process

A fully characterized bivariate normal distribution only requires marginal normal distribution mean and variance parameters with the correlation coefficient for full mathematical description. The correlation coefficient determines the correlation between both the OMT and OBRT random variables. Mathematically, the bivariate normal probability density function which describes the mission process is expressed, as follows,

$$P(t^{(b)}, t^{(m)}) = \frac{1}{2\pi\sigma_{RT}\sigma_{MT}\sqrt{1-\rho^2}} \exp\left[-\frac{Q(t^{(b)}, t^{(m)})}{2(1-\rho^2)}\right], \quad (5.9)$$

where σ_{RT} is the standard deviation of the overall battery run-time, σ_{MT} is the standard deviation of the overall mission time, and ρ is the correlation coefficient between the OMT and OBRT processes. Additionally, $t^{(b)}$ and $t^{(m)}$ are the independent variables of time for the battery run-time and mission time variables, respectively. The term in the exponential function is defined in the following manner,

$$Q(t^{(b)}, t^{(m)}) = \frac{(t^{(b)} - \mu_{RT})^2}{\sigma_{RT}^2} + \frac{(t^{(m)} - \mu_{MT})^2}{\sigma_{MT}^2} - 2\rho \frac{(t^{(b)} - \mu_{RT})(t^{(m)} - \mu_{MT})}{\sigma_{MT}\sigma_{RT}}, \quad (5.10)$$

where μ_{MT} and μ_{RT} are the mean overall mission time and overall battery run-time, respectively. Displayed in Figure 5.9 is the reliability failure surface where OMT > OBRT. As introduced in Section 2.4, PoMC is the volume of the bivariate distribution that lies beyond the failure surface.

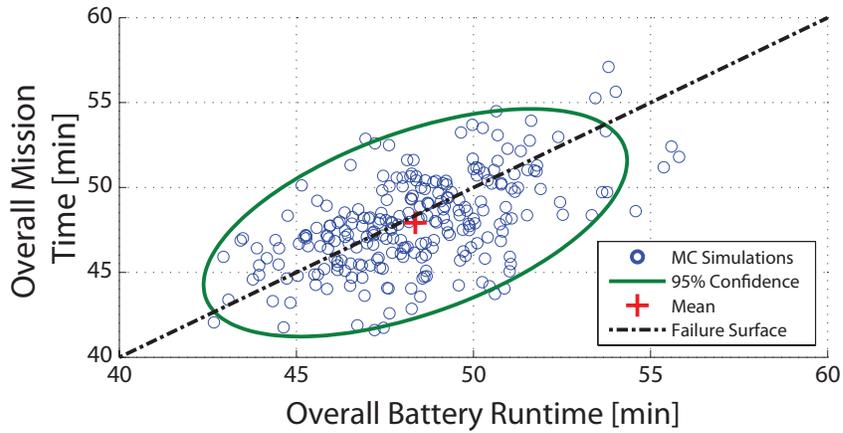


Figure 5.9: Bivariate normal characterization of the Monte Carlo UGV study results.

5.2.3 Online Bayesian Estimation of Correlation

Each individual OMT and OBRT prediction yields the marginal distributions of the overall mission bivariate distribution. As a result, the mission process correlation must be estimated via the OBRT/OMT prediction pairs $(\hat{\mu}_{MT}, \hat{\mu}_{RT})$. Furthermore, during the initial deployment of a mobile system, few $(\hat{\mu}_{MT}, \hat{\mu}_{RT})$ prediction pairs are available. As such, a recursive Bayesian updating scheme was developed to incorporate each successive prediction pair for process correlation estimation. This process is summarized in Figure 5.10. With additional predictions, the uncertainty associated with the estimated correlation coefficient should further diminish [64].

The posterior density function is proportional to the product of the likelihood function and the chosen prior distribution. Mathematically, the

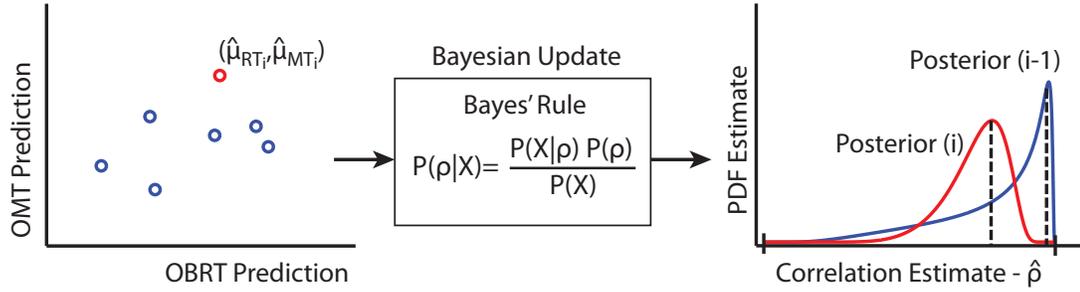


Figure 5.10: Online correlation estimation scheme utilizing Bayesian inference with marginal predictions of OMT and OBRT.

posterior density for correlation is expressed as,

$$P(\rho|\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}}) \propto P(\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}}|\rho) P(\rho), \quad (5.11)$$

where $\hat{\mu}_{RT_{1:k}}$ and $\hat{\mu}_{MT_{1:k}}$ are the first through k 'th mean predictions of OBRT and OMT, respectively forming the $(OBRT_i, OMT_i)$ prediction pair. Furthermore, $P(\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}}|\rho)$ is the likelihood function. The likelihood function returns the probability of the observed predicted pairs conditioned on a given correlation coefficient. Finally, $P(\rho)$ is the prior density function for correlation.

As the selected prior distribution contributes to the posterior estimate, careful selection remains necessary to ensure the prior does not bias the posterior density estimate. A common minimally informative prior distribution scheme, which is based on Fisher information, is the Jeffery's prior, shown in Figure 5.11. Given known marginal variances, the Jeffery's prior for the correlation coefficient of a bivariate normal distribution is given [117], as follows,

$$P(\rho) \propto \hat{\sigma}_{RT} \hat{\sigma}_{MT} (1 - \rho^2)^{-\frac{3}{2}}. \quad (5.12)$$

where $\hat{\sigma}_{RT}$ and $\hat{\sigma}_{MT}$ are the most recently predicted standard deviations for OBRT and OMT, respectively.

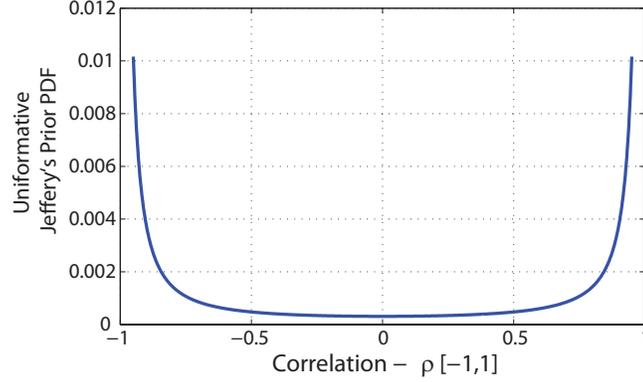


Figure 5.11: Jeffery's uniformative prior for Bayesian inference of correlation with known marginals.

The likelihood function takes the bivariate normal formulation given in equation (5.9). However, provided multiple independent predictions, the resulting total likelihood of the prediction set, $(\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}})$, is given by the product of each individual observation [64]. It is important to note that when not explicitly denoted as a vector of predictions, the variance or mean values expressed in the notation represent the most recent prediction. For example, $\hat{\mu}_{RT_{1:k}}$ indicates the vector of OBRT mean predictions, whereas $\hat{\mu}_{RT}$ indicates the k'th OBRT mean prediction only. Resultantly, the prediction likelihood function is given by,

$$P(\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}} | \rho) = \left[2\pi \hat{\sigma}_{RT} \hat{\sigma}_{MT} \sqrt{1 - \rho^2} \right]^{-k} \exp \left[-\frac{Q_{1:k}^{(rt)} + Q_{1:k}^{(mt)} + 2\rho Q_{1:k}^{(c)}}{2(1 - \rho^2)} \right]. \quad (5.13)$$

where k is the number of prediction pairs. Furthermore, the summation terms

in the exponential are defined as follows.

$$Q_{1:k}^{(rt)} = \sum_{i=1}^k \frac{(\hat{\mu}_{RT_i} - \hat{\mu}_{RT})^2}{\hat{\sigma}_{RT}^2} \quad (5.14)$$

$$Q_{1:k}^{(mt)} = \sum_{i=1}^k \frac{(\hat{\mu}_{MT_i} - \hat{\mu}_{MT})^2}{\hat{\sigma}_{MT}^2} \quad (5.15)$$

$$Q_{1:k}^{(c)} = \sum_{i=1}^k \frac{(\hat{\mu}_{RT_i} - \hat{\mu}_{RT})(\hat{\mu}_{MT_i} - \hat{\mu}_{MT})}{\hat{\sigma}_{RT}\hat{\sigma}_{MT}} \quad (5.16)$$

The product of the prior distribution in equation (5.12) and the likelihood function in equation (5.13) yield the posterior distribution for ρ via the Bayesian inference of equation (5.11). After algebraic simplifications, the posterior distribution is found to be proportional to the following expression,

$$P(\rho|\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}}) \propto \left[\sqrt{1 - \rho^2} \right]^{-\frac{1}{2}(k+3)} \exp \left[-\frac{Q_{1:k}^{(rt)} + Q_{1:k}^{(mt)} - 2\rho Q_{1:k}^{(c)}}{2(1 - \rho^2)} \right]. \quad (5.17)$$

Implementation of equation (5.17) with a single instance of the Monte Carlo simulations from Section 5.2.2 to estimate the probability distributions of $\hat{\rho}$ is shown in Figure 5.12.

To demonstrate the statistical efficacy of the Bayesian updating scheme for correlation estimation, the methodology was applied sequentially to the Monte Carlo simulation results given in Figure 5.8. Beginning with a single sample, Monte carlo realizations were added sequentially to equation (5.17) as predicted mean pairs $(\hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}})$. The maximum *a posteriori* (MAP) metric was chosen for the estimate of ρ [64]. Succinctly stated, the MAP

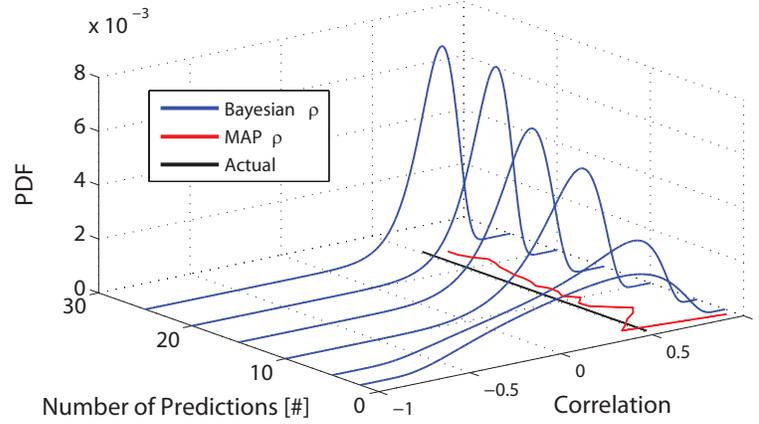


Figure 5.12: Single Bayesian estimation of OBRT and OMT correlation via sequential updating using single set of simulation data.

estimate of correlation is,

$$\hat{\rho}_k = \arg \max_{\rho} P(\rho | \hat{\mu}_{RT_{1:k}}, \hat{\mu}_{MT_{1:k}}), \quad (5.18)$$

where $\hat{\rho}_k$ is the MAP estimate of correlation given k predictions. As seen in Figure 5.13, the MAP estimate converges to the actual correlation coefficient in under five predictions. Figure 5.13 illustrates the properties of the Bayesian correlation estimate over 500 realizations of the mission process. Subsequently, the confidence interval constricts given additional prediction evidence.

5.2.4 Assessment of Mission Reliability

Combination of predictions of the marginal statistics and the estimate of correlation produce the bivariate normal approximation of the mission process given by equation (5.9). A mission remains feasible contingent on the battery run-time exceeding the required time for tasks and locomotion. Con-

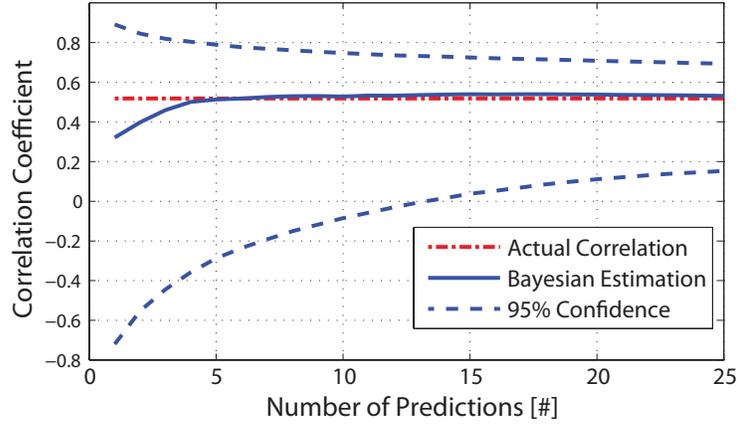


Figure 5.13: Mean Bayesian estimation of OBRT and OMT correlation via sequential updating using MC simulation data.

versely, infeasibility arises when mission time exceeds battery run-time. To compute the probability of infeasibility, or $P(OMT > OBRT)$, the volume under the bivariate normal in the region of $t^{(m)} > t^{(b)}$ must be found. As such, the region of infeasibility is bounded below by the $t^{(m)} = t^{(b)}$ line.

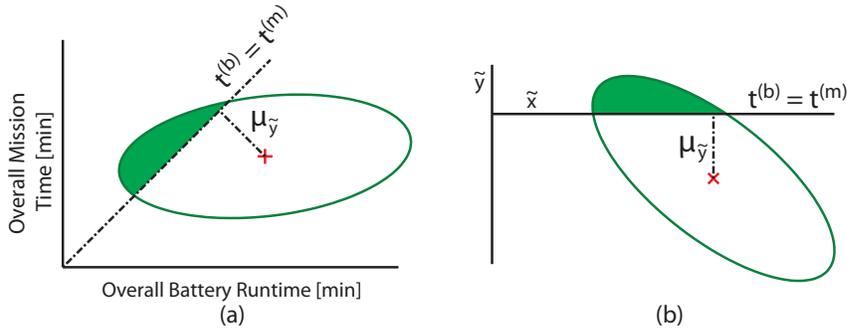


Figure 5.14: Rotation/translations of mission random variable for mission probability computational efficiency.

Rather than the computationally intense numerical integration of the untransformed bivariate distribution given in equation (5.9), rotation/translation

manipulations simplify the probability calculation. Applying a 45 degree rotation transformation and zeroing translation to the mission process converts the $t^{(m)} = t^{(b)}$ line into an augmented x-axis which simplifies numerical integration, as seen in Figure 5.14. Expressing the bivariate normal distribution of equation (5.9) in vector form yields,

$$P(t^{(b)}, t^{(m)}) = \frac{1}{2\pi |\Sigma|} \exp \left[-\frac{1}{2} (\mathbf{t} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{t} - \boldsymbol{\mu}) \right] \quad (5.19)$$

where $\mathbf{t} = [t^{(b)} \quad t^{(m)}]^T$, $\boldsymbol{\mu} = [\hat{\sigma}_{RT} \quad \hat{\sigma}_{MT}]^T$, and $|\Sigma|$ is the determinate of the covariance matrix, where the covariance is expressed as,

$$\Sigma = \begin{bmatrix} \hat{\sigma}_{RT}^2 & \hat{\rho}_k \hat{\sigma}_{RT} \hat{\sigma}_{MT} \\ \hat{\rho}_k \hat{\sigma}_{RT} \hat{\sigma}_{MT} & \hat{\sigma}_{MT}^2 \end{bmatrix}. \quad (5.20)$$

Normalization of the mission means illustrated in Figure 5.14 requires finding the minimum distance between the process mean and the $t^{(m)} = t^{(b)}$ line. In vector notation, the augmented mean is expressed as,

$$\tilde{\boldsymbol{\mu}} = \left[0 \quad \frac{1}{\sqrt{2}} (\hat{\mu}_{RT} - \hat{\mu}_{MT}) \right]^T. \quad (5.21)$$

Applying a univariate rotational transformation, $\tilde{\Sigma} = R^{-1} \Sigma R$, with a 45 degree rotation matrix results in the following augmented covariance matrix,

$$\tilde{\Sigma} = \frac{1}{2} \begin{bmatrix} \hat{\sigma}_{RT}^2 + 2\hat{\rho}_k \hat{\sigma}_{RT} \hat{\sigma}_{MT} + \hat{\sigma}_{MT}^2 & \hat{\sigma}_{MT}^2 - \hat{\sigma}_{RT}^2 \\ \hat{\sigma}_{MT}^2 - \hat{\sigma}_{RT}^2 & \hat{\sigma}_{RT}^2 - 2\hat{\rho}_k \hat{\sigma}_{RT} \hat{\sigma}_{MT} + \hat{\sigma}_{MT}^2 \end{bmatrix}. \quad (5.22)$$

Integration of the region bounded below by the $t^{(m)} = t^{(b)}$ line in augmented notation with $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$ is expressed as,

$$P(OMT > OBRT) = \iint_{(\tilde{x}, \tilde{y}) : \tilde{y} > 0} \frac{1}{2\pi |\tilde{\Sigma}|} \exp \left[-\frac{1}{2} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^T \tilde{\Sigma}^{-1} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}}) \right] d\tilde{x} d\tilde{y}, \quad (5.23)$$

where $\tilde{\mathbf{x}} = [\tilde{x} \ \tilde{y}]^T$. As a result of variable dependence elimination, integration of the above expression only requires numerical considerations in the augmented first and second cartesian quadrants. For implementation, Riemann sum numerical integration is utilized to compute the integral in equation (5.23).

5.2.5 Online Restructuring of Mission Objectives

Integration of equation (5.23) yields the currently selected mission probability of success contingent on remaining battery energy. However, in the event of low probability of completion of a primary mission, a contingency mission could be automatically adopted by a supervisory control algorithm or recommended to a vehicle teleoperator. In the following example, a mobile robot must travel to and accomplish three prespecified tasks, as seen in Figure 5.15.

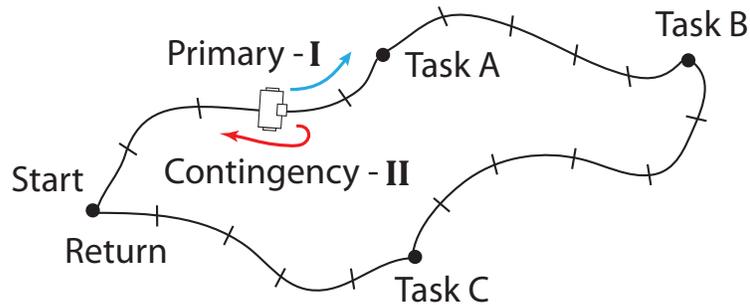


Figure 5.15: Example *a priori* specified mission with three tasks and two potential mission plans: the desired mission and the contingency plan.

Before deployment of the mobile robotic system, the desired mission plan is developed including the potential tasks and drive paths. This *a priori*

specified plan is the primary mission path. Contrastingly, given any component failures or insufficient onboard energy, a contingency return plan is also specified. Provided implementation of the online PoMC assessment algorithm, the probability of $OMT > OBRT$ can be found to evaluate the primary mission viability. If the primary mission probability falls below a desired mission success threshold, the contingency plan is adopted and the primary mission canceled, as seen in Figure 5.16.

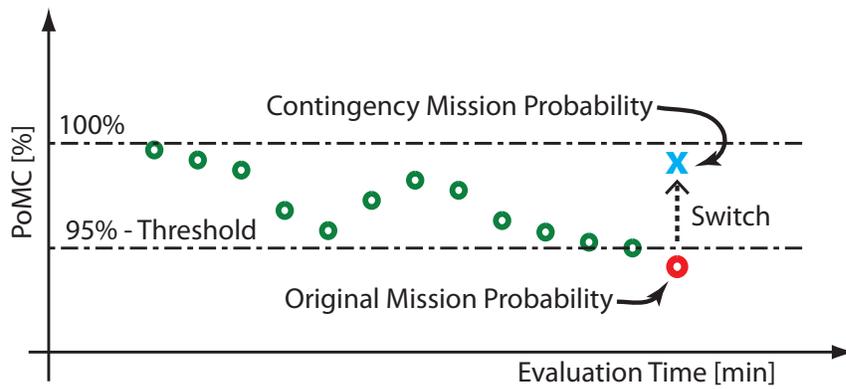


Figure 5.16: Online restructuring of mission plan to adopt a contingency plan via PoMC decision metric.

5.3 Experimental/Field Validation

To validate the efficacy of the proposed PoMC estimation algorithm, the methodology was implemented with data from experimental studies utilizing a mobile system in unstructured terrain. Again, the unmanned ground vehicle was chosen for validation of the methodology as the loading profiles experienced during field operation remain stochastic and exhibit large transient

jumps. Firstly, using the DaNI UGV discharge studies discussed in detail in Section 4.4.2, the PoMC was evaluated for a range of desired mission distances. The PoMC of the DaNI UGV studies is discussed in Section 5.3.1.

5.3.1 In-house Small Ground Vehicle Test Stand Study

As introduced in Section 4.4.2, twenty-two discharge studies were conducted with the DaNI UGV on the stochastic terrain environment. In addition to the power loads and battery discharge characteristics, the velocities of the UGV and the cumulative distance traveled were recorded by a fixed LIDAR system attached to the terrain turntable. The velocity characteristics of each discharge test are discussed in Appendix B. The PoMC estimation methodology was utilized to compute the mission probability for each discharge test with different desired mission distances. As a comparison, the PoMC estimates are also computed via the conjectured method of [21] where the OMT and OBRT process are presumed independent. The fidelity of the PoMC predictions are evaluated via the CRA metric, introduced in Section 2.1.4

5.3.1.1 Experimental UGV Mission Process

As introduced in Section 4.4.2, the DaNI UGV traversed the stochastic terrain environment until battery shutdown. Consequentially, each DaNI UGV experiment serves as a realization of a mission process, akin to Figure 5.1. Furthermore, as the vehicle was confined to a limited operation space, the mission distance of the DaNI UGV is given by the cumulative longitudi-

nal displacement, measured via the fixed LIDAR system. The total distance traversed by for each UGV experiment as given in Figure 5.17.

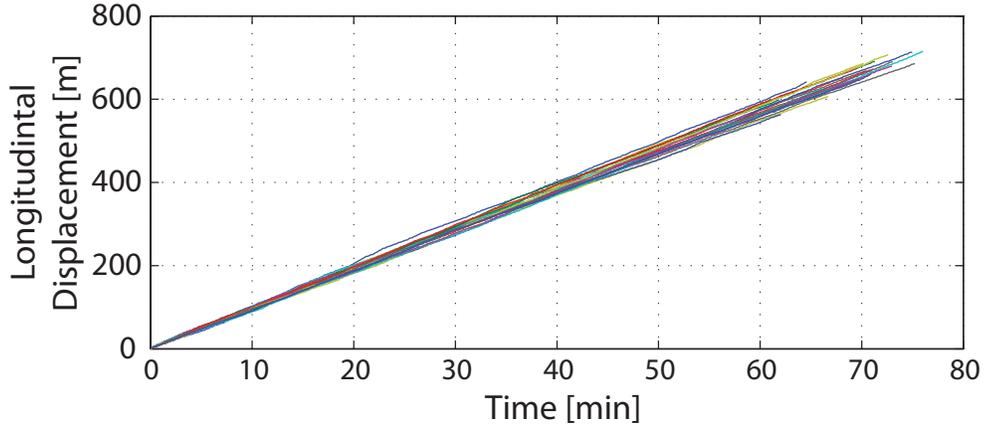


Figure 5.17: Cumulative longitudinal displacement of the DaNI UGV for all inhouse discharge studies.

Velocity of the DaNI UGV depended on the issued rate command and the extant terrain properties. Resultantly, the vehicle velocity demonstrated marked stochasticity, as illustrated in Figure 5.18. The experiment run times (ground truth OBRT) ranged from 61.72 to 76.03 minutes, whereas the cumulative distance traveled ranged from 562.95 to 715.27 meters. Each mission process, with the ground truth OBRT and mission distance, is illustrated in Figure 5.19.

While the Monte Carlo UGV simulation studies demonstrated bivariate normality, similar analysis was conducted to ascertain if a more general process is remains approximated via a bivariate normal distribution. Applying the Mardia normality test, see Section 5.2.2, with a 5% significance level for both

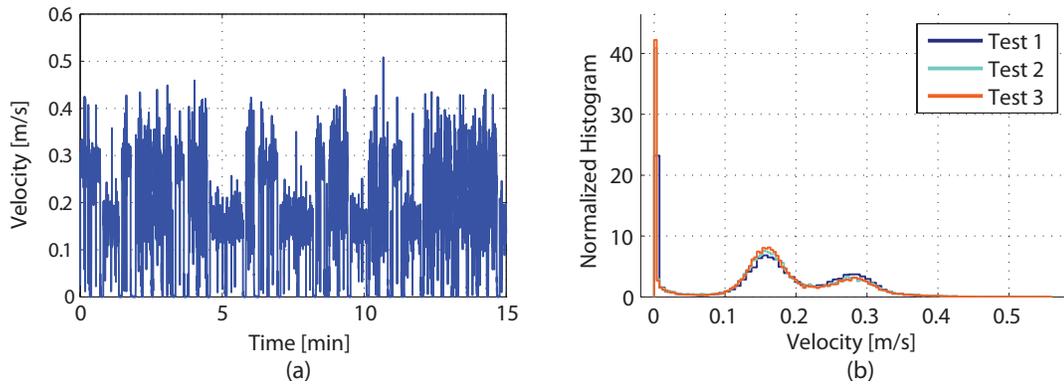


Figure 5.18: (a) Measured vehicle velocity of the DaNI UGV via differential LIDAR. (b) Normalized histogram depicting the non-Gaussian velocity trends of the first three experiments.

the skewness and kurtosis, the null hypothesis of normality is not rejected. More concretely, the 5% significant statistic for skewness for the experimental data is 9.49 which is not exceeded by the Mardia skewness statistic of 1.21. Furthermore, the 5% significant statistic for kurtosis is 1.64 which remains greater than the Mardia kurtosis statistic of 1.15. Resultantly, the experimental test data can be approximated as a bivariate normal mission process akin to the theory discussed in Section 5.2.2.

5.3.1.2 Probability of Mission Completion Experimental Analysis

To assess the performance of the proposed PoMC algorithm, the UGV DaNI studies are viewed as multiple realizations of a UGV mission deployment. For an actual online implementation of the PoMC algorithm, the desired mission distance, D_m , would be specified *a priori*. However, for analysis of PoMC prediction fidelity, a range of mission profiles are explored. Illustrated in Fig-

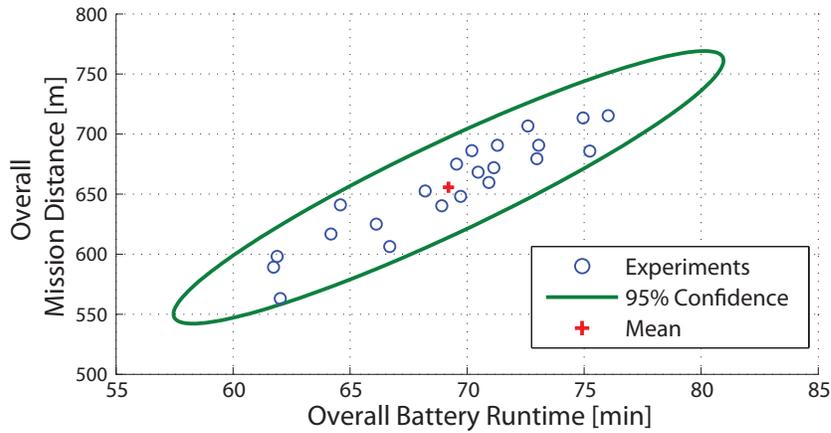


Figure 5.19: Experimental results from UGV terrain discharge tests comparing discharge time and cumulative distance traveled with a bivariate normal distribution fit.

Figure 5.20 are a range of mission profiles ($D_m = [500, 700]$ meters) using the DaNI UGV data. If a particular experiment failed prior to the specified mission distance, the mean velocity was used to predict the OMT as if the system continued operating. Also displayed is the failure surface ($OMT > OBRT$). As D_m increases, the PoMC, as expected, falls.

For an individual realization of the mission process (experiment # 4), the predicted PoMC is shown in Figure 5.21 for both $D_m = 550$ m and $D_m = 650$ m. The mission feasibility prediction bounds result from the uncertainty associated with the correlation coefficient, shown in Figure 5.13. For the $D_m = 550$ m mission, the ground truth PoMC is 0.99. Correspondingly, the PoMC prediction made over the mission run-time indicate a 95% confidence of a PoMC greater than 0.83. Contrastingly, for the $D_m = 650$ m mission with a ground truth PoMC of 0.55, the prediction indicates at the first time

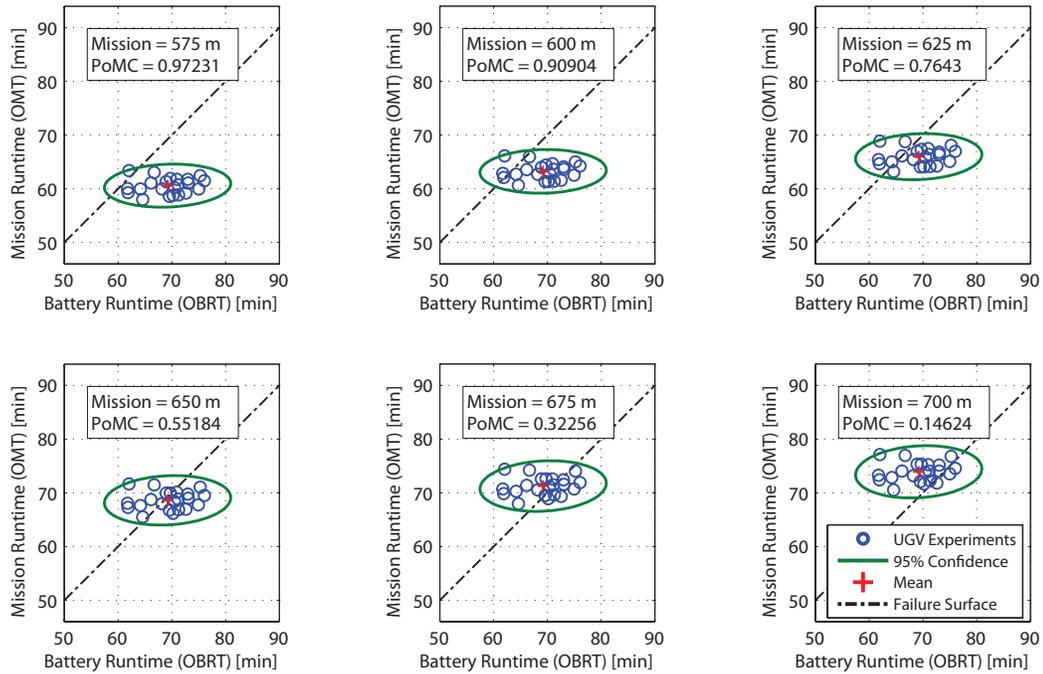


Figure 5.20: Ground truth probability of mission completion for a range (575m to 700m) mission profiles for the DaNI UGV.

of prediction ($t^c = 5$ minutes) a PoMC of 0.51 ± 0.09 . While process noise results in PoMC prediction fluctuations, the mean prediction over the prediction horizon is 0.54. Consequentially, the UGV controller would be informed to adopt the contingency plan.

To evaluate the PoMC prediction fidelity over all available experimental DaNI UGV data sets, the mean of the PoMC predictions were compared. As the predictions should accurately assess the ground truth PoMC, the mean of the PoMC prediction should converge to the ground truth PoMC for each particular mission profile, D_m . For each DaNI UGV test, the PoMC predictor made predictions at time 5-40 minutes for D_m ranging from 475 m (ground

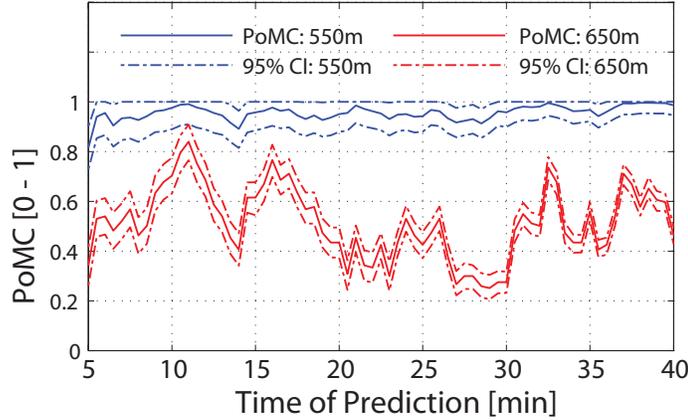


Figure 5.21: Online predictions of PoMC for experiment number four given desired mission distances of 550m and 650m over the UGV run-time.

truth $\text{PoMC} = 1$) to 775 m (ground truth $\text{PoMC} = 0$). The average of the twenty two PoMC predictions over the experiment run-time is shown in Figure 5.22. The average PoMC for all UGV studies converges to the expected value of the ground truth PoMC from Figure 5.21 for all D_m profiles. The offset of the prediction for $D_m = 675$ is conjectured to result from the bivariate approximation of the mission process. As the marginal distribution of the OBRT is non-normal, the PoMC predictions are only approximate. Planned extensions of the methodology to include reliability copulas which handle non-normal distributions are discussed in the conclusions.

As discussed in Section 2.4, recent proposed PoMC algorithms for mobile systems discount the correlation between the OMT and OBRT [21]. Contrastingly, available onboard energy in battery systems depends on the operation characteristics of the mission. As such, process correlation is necessary for accurate PoMC prediction. To illustrate the necessity of correlation for PoMC

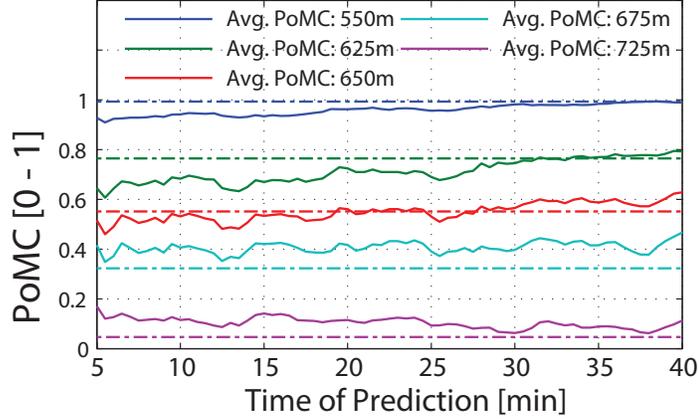


Figure 5.22: Average PoMC for all twenty-two experiments compared to actual mission probability rates (dashed line) for numerous desired missions.

prediction, the proposed PoMC method was compared to the assumption of independence suggested in [21],

$$\hat{P}_I(OMT > OBRT) = \int_{-\infty}^{\infty} \Phi_{OMT}(t) \phi_{OBRT}(t) dt \quad (5.24)$$

where $\Phi_{OMT}(t)$ is the normal CDF of OMT and $\phi_{OBRT}(t)$ is the normal PDF of OBRT.

The analysis conducted to generate Figure 5.22 was repeated with both the proposed correlated PoMC method and equation (5.24). The PoMC, using each method, was predicted at times ranging from 5 minutes to 40 minutes for each mission profile. Displayed in Figure 5.23 are the predicted probabilities using each method for given D_m mission profiles. The error bars indicate the CRA standard deviation for the twenty two UGV experiments.

Clearly visible in Figure 5.23 is the lack of precision of the indepen-

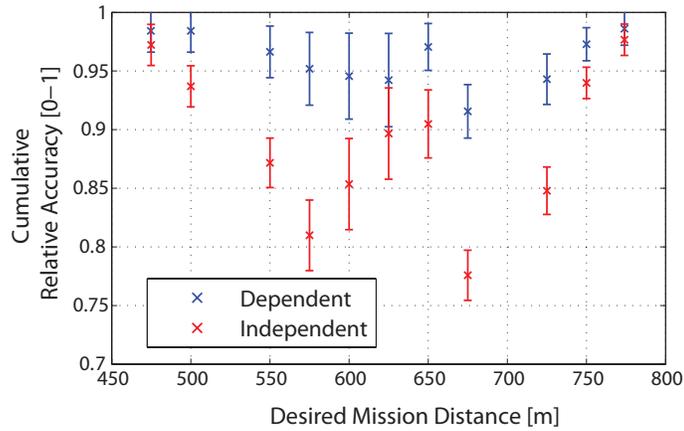


Figure 5.23: Comparison of the CRA for PoMC prediction using the Bayesian $\hat{\rho}$ estimates and assuming no process correlation ($\rho = 0$) for difference desired mission profiles.

dent PoMC prediction scheme. When the mission ground truth PoMC is certainly either failure (PoMC = 0) or success (PoMC = 1), predictions from both methods maintain high CRA. However, as expected, the independent scheme fails due to the underestimation/overestimation of PoMC. Near the 50% PoMC mission distance, the prediction errors of the dependent scheme recover. This recovery results from the independent PoMC predictor transitioning from overestimating to underestimating the PoMC. Furthermore, the prediction fidelity of each scheme falls in the mission range between 675 and 725 meters. As mentioned, the conjecture is that prediction accuracy falls as a result of the bivariate normal approximation. Future work on improving PoMC is addressed in the conclusions. Finally, a box plot, in Figure 5.24, summarizes all the CRA predictions made by both the proposed bivariate PoMC algorithm and the independent assumption.

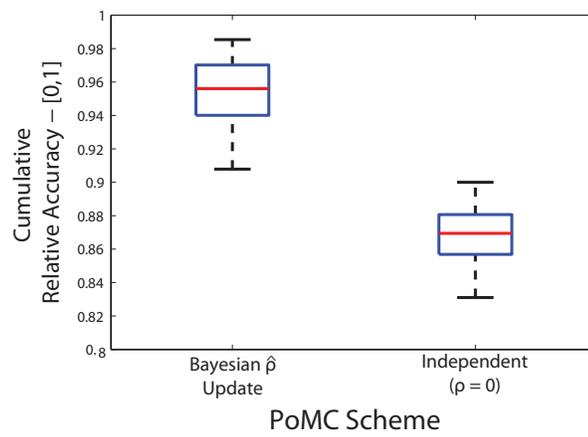


Figure 5.24: Comparison of the CRA of the Bayesian and independent prediction schemes for all DaNI UGV experiments (where $0 < GTPoMC < 1$).

Chapter 6

Energy-Aware Control

As has been emphasized in this work, mobile system shutdown frequently results from transient power demands imposed by unstructured environments as opposed to state-of-charge depletion. In the UGV experimental studies presented, transient current loads resulted in system shutdown of the Packbot and 81.8 % of the indoor/outdoor DaNI UGV tests. To mitigate these transient shutdown conditions and extend mobile system life, a model predictive control (MPC) scheme has been derived which sequentially optimizes input commands for a UGV to maintain feasible battery terminal voltages. Furthermore, the MPC scheme uses the process model to ensure feasibility over an entire prediction interval.

As introduced in Section 2.5, existing methodologies require *a priori* knowledge of loads and all methods found in the literature do not account for transient shutdown conditions. As such, in this chapter, preliminary work on an energy-aware control scheme for online command regulation is presented that mitigates transient shutdown conditions to extend system run-time. With recent advances in the computational speed of optimization routines, model predictive control, which accounts for hard constraints, was selected for online

energy-aware control for mobile systems.

6.1 Energy-aware model predictive control

Model predictive control is a receding horizon control methodology that has seen rapid increase in utilization in the past decade [119, 159]. The increase has resulted from the development of efficient quadratic programming optimization solvers which can be implemented online on a mobile system, such as a UGV [25, 126]. The goal of the MPC presented in this work is to track a desired command trajectory and ensure the satisfaction of the battery voltage constraint. If the issued command will result in system shutdown within the prediction horizon, the MPC methodology should adjust accordingly to prevent system shutdown. Commands for the mobile system are assumed to be issued either via a vehicle teleoperator or a supervisory control algorithm.

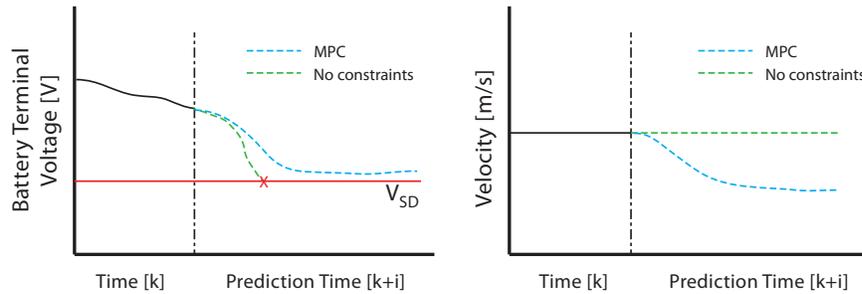


Figure 6.1: Illustration of the energy-aware model predictive control scheme to prevent transient shutdown conditions.

For implementation of the receding horizon scheme, at each discrete time step, k , an optimization routine is implemented over the control horizon, N . This optimization minimizes a cost function via selection of an optimal

control input vector, $u^*(k : k + N)$. Furthermore, the MPC optimization also provides for constraints on states, inputs and outputs, which restrict the feasible set space of the optimization [160]. Provided feasibility of the constraint set, the resulting $u^*(k : k + N)$ provides the optimal control vector over the control horizon. A general cost function for state regulation with a final state cost is expressed as follows,

$$J_k = \sum_{i=1}^N x_{k+i}^T Q_i x_{k+i} + u_{k+i-1}^T R_i u_{k+i-1} + x_L^T S x_L \quad (6.1)$$

where Q_i is the state weight matrix, R_i is the control weight matrix, and S is the final state weight matrix. Additionally, MPC can include constraints, as follows,

$$\begin{aligned} \underline{x}_k &\leq x_k \leq \bar{x}_k \\ \underline{u}_k &\leq u_k \leq \bar{u}_k \\ \underline{y}_k &\leq y_k \leq \bar{y}_k \end{aligned} \quad (6.2)$$

where \underline{x}_k , \underline{u}_k , and \underline{y}_k are the minimum state, control and output values at time k , respectively. Furthermore, \bar{x}_k , \bar{u}_k , and \bar{y}_k are the maximum state, control and output values. In the MPC scheme at each time step, only the first optimized value, $u^*(k)$ is input into the system. Schematically, the receding horizon control scheme is illustrated in Figure 6.2. While a standard quadratic programming formulation is adopted for the energy-aware control scheme, the nonlinear MPC methodology is presented as follows for clarity of notation [161].

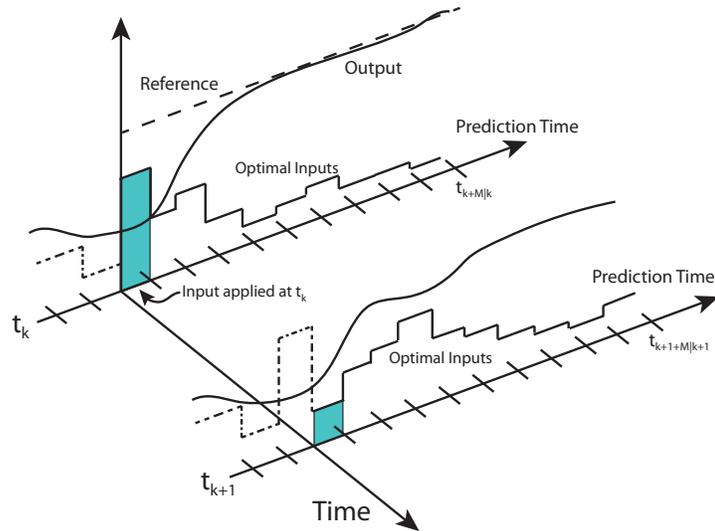


Figure 6.2: Schematic depiction of the receding horizon control scheme for two steps in time.

6.1.1 Model Formulation and Nomenclature

For online MPC implementation, the optimization routine to determine the control input must be computationally efficient and solve for control inputs at a rate faster than the system dynamics [159]. Historically, computational limitations have restricted the MPC methodology to process control applications with slow transients [162]. However, structuring of the MPC optimization as a quadratic program (QP) allows for use of efficient QP solvers such as active set and interior point methods [160]. For linear time-invariant systems, the QP formulation of MPC has been implemented with success in numerous applications [163, 164]. However, the QP, by definition, requires a linear dynamic model of the process to ensure convexity [165]. To extend the computationally efficient QP routine to a nonlinear process, such as the

battery/mobile system dynamics, a suboptimal time-varying MPC approach has been adopted and implemented [160, 161]. The following discussion details the suboptimal time-varying MPC approach for nonlinear system control with notation following [161]. For considerations of MPC stability, which are not discussed in this work, see [166].

A discrete-time nonlinear system and nonlinear output equation are generally represented in state-space form as follows,

$$x_{k+1} = f(x_k, u_k, d_k) \quad (6.3)$$

$$y_{k+1} = h(x_{k+1}, u_{k+1}, d_{k+1}) \quad (6.4)$$

where x_k are the model states, u_k are the model inputs, d_k are model disturbances which are assumed to be known at time k . As the nonlinear dynamics are not admissible in the QP formulation of MPC, the nonlinear dynamics can be linearized over a trajectory of the MPC horizon, N . Provided a reference control input trajectory, $\mathbf{u}^0 = [u_k, u_{k+1}, \dots, u_{k+N}]$, disturbance trajectory, $\mathbf{d}^0 = [d_k, d_{k+1}, \dots, d_{k+N}]$, and an initial state, x_k , nominal state and output trajectories of the nonlinear system can be computed via equation (6.3). Resultantly, a nominal state trajectory, $\mathbf{x}^0 = [x_k, x_{k+1}, \dots, x_{k+N}]$, and nominal output trajectory, $\mathbf{y}^0 = [y_k, y_{k+1}, \dots, y_{k+N}]$ are obtained.

Linearization of equations (6.3) over the nominal trajectory yield linearized state matrices,

$$A_k = \left. \frac{\partial f}{\partial x} \right|_{\xi_k = \xi_k^0} \quad B_k = \left. \frac{\partial f}{\partial u} \right|_{\xi_k = \xi_k^0} \quad E_k = \left. \frac{\partial f}{\partial d} \right|_{\xi_k = \xi_k^0} \quad (6.5)$$

where the notation $\xi_k = \{x_k, u_k, d_k\}$ and $\xi_k^0 = \{x_k^0, u_k^0, d_k^0\}$ has been adopted for clarity. Furthermore, the nonlinear output equations (6.3) similarly yields,

$$C_k = \left. \frac{\partial h}{\partial x} \right|_{\xi_k = \xi_k^0} \quad D_k = \left. \frac{\partial h}{\partial u} \right|_{\xi_k = \xi_k^0} \quad F_k = \left. \frac{\partial h}{\partial d} \right|_{\xi_k = \xi_k^0} \quad (6.6)$$

With the linearized states matrices defined above, a linear time-varying perturbation model can be formulated that approximates equations (6.3) [167]. As a perturbation model, the linear time-varying model approximates the variations from the nominal trajectory. As such, the state, input, output and known disturbance variations are given by, $\delta x_k = x_k - x_k^0$, $\delta u_k = u_k - u_k^0$, $\delta y_k = y_k - y_k^0$, and $\delta d_k = d_k - d_k^0$, respectively. With these perturbation parameters, the approximate state and output time-varying equations become,

$$\delta x_{k+1} = A_k \delta x_k + B_k \delta u_k + E_k \delta d_k \quad (6.7)$$

$$\delta y_k = C_k \delta x_k + D_k \delta u_k + F_k \delta d_k \quad (6.8)$$

For clarity of notation, the perturbation variables over the MPC prediction horizon, N , are expressed using bold notation. Resultantly, $\delta \mathbf{x} = [\delta x_k, \delta x_{k+1}, \dots, \delta x_{k+N}]$, $\delta \mathbf{u} = [\delta u_k, \delta u_{k+1}, \dots, \delta u_{k+N}]$, $\delta \mathbf{y} = [\delta y_k, \delta y_{k+1}, \dots, \delta y_{k+N}]$, and $\delta \mathbf{d} = [\delta d_k, \delta d_{k+1}, \dots, \delta d_{k+N}]$.

6.1.2 Model Predictive Control with a Time-Varying Linear Perturbation Model

For the linear time-varying system presented, the general cost function in equation (6.3) can be expressed in terms of the perturbation variables. The

cost function in vector notation is expressed as follows,

$$J_k = \delta \mathbf{x}^T \hat{Q} \delta \mathbf{x} + \delta \mathbf{u}^T \hat{R} \delta \mathbf{u} \quad (6.9)$$

where \hat{Q} and \hat{R} the state and control cost matrices for each state and control vector, respectively. Concretely, these matrices are given as follows,

$$\hat{Q} = \begin{bmatrix} Q_1 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & Q_{N-1} & 0 \\ 0 & \cdots & 0 & S \end{bmatrix} \quad \hat{R} = \begin{bmatrix} R_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & R_N \end{bmatrix} \quad (6.10)$$

To formulate the MPC objective function in QP optimization form, the state variables, $\delta \mathbf{x}$ must be implicitly removed from the cost function. As a result of the linear dynamics, however, the future state vectors can be computed via only the input vector, $\delta \mathbf{u}$ and the initial state δx_k [161]. Solving recursively for the states, the augmented time-varying state equation, requiring only inputs and the initial condition, is given as follows,

$$\delta \mathbf{x} = \hat{A} \delta x_k + \hat{B} \delta \mathbf{u} + \hat{E} \delta \mathbf{d} \quad (6.11)$$

where the \hat{A} , \hat{B} , \hat{E} matrices are found recursively as follows,

$$\hat{A} = [A_k \quad A_{k+1} A_k \quad \cdots \quad A_{k+N} A_{k+N-1} \cdots A_k]^T \quad (6.12)$$

$$\hat{B} = \begin{bmatrix} B_k & 0 & \cdots & 0 \\ A_{k+1} B_k & B_{k+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ A_{k+N} \cdots A_{k+1} B_k & A_{k+N} \cdots A_{k+1} B_{k+1} & \cdots & B_{k+N} \end{bmatrix} \quad (6.13)$$

$$\hat{E} = \begin{bmatrix} E_k & 0 & \cdots & 0 \\ A_{k+1} E_k & E_{k+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ A_{k+N} \cdots A_{k+1} E_k & A_{k+N} \cdots A_{k+1} E_{k+1} & \cdots & E_{k+N} \end{bmatrix} \quad (6.14)$$

Equally, the perturbation output equation can be expressed in terms of only the input vector and the perturbation state initial conditions at time k . As such, the output equation is given as,

$$\delta \mathbf{y} = \hat{C} \delta x_k + \hat{D} \delta \mathbf{u} + \hat{F} \delta \mathbf{d} \quad (6.15)$$

where the recursively found \hat{C} , \hat{D} , and \hat{F} , matrices are expressed as follows,

$$\hat{C} = [C_k \quad C_{k+1}A_k \quad \cdots \quad C_{k+N}A_{k+N-1} \cdots A_k]^T \quad (6.16)$$

$$\hat{D} = \begin{bmatrix} D_k & 0 & 0 & \cdots & 0 \\ C_{k+1}B_k & D_{k+1} & 0 & 0 & 0 \\ C_{k+1}A_{k+1}B_k & C_{k+2}B_{k+1} & \ddots & 0 & 0 \\ \vdots & \vdots & \cdots & D_{k+N-1} & 0 \\ C_{k+N}A_{k+N-1} \cdots A_{k+1}B_k & \cdots & \cdots & C_{k+N}B_{k-N-1} & D_{k+N} \end{bmatrix} \quad (6.17)$$

$$\hat{F} = \begin{bmatrix} F_k & 0 & 0 & \cdots & 0 \\ C_{k+1}B_k & F_{k+1} & 0 & 0 & 0 \\ C_{k+1}A_{k+1}B_k & C_{k+2}B_{k+1} & \ddots & 0 & 0 \\ \vdots & \vdots & \cdots & F_{k+N-1} & 0 \\ C_{k+N}A_{k+N-1} \cdots A_{k+1}B_k & \cdots & \cdots & C_{k+N}B_{k-N-1} & F_{k+N} \end{bmatrix} \quad (6.18)$$

Substitution of equation (6.11) into equation (6.9) yields a QP-friendly MPC cost function where the only variable for optimization is $\delta \mathbf{u}$.

$$J_k = \delta \mathbf{u}^T [\hat{B}^T \hat{Q} \hat{B} + \hat{R}] \delta \mathbf{u} + 2 [\hat{A} \delta x_k + \hat{E} \delta \mathbf{d}]^T \hat{Q} \hat{B} \delta \mathbf{u} \quad (6.19)$$

For the energy-aware MPC scheme, the QP optimization must also include both input constraints (constraints on the PWM duty cycle) and output constraints (the battery terminal voltage). Akin to the original optimization

objective in equation (6.3), the QP constraints must be expressed in the form of equation (6.2). As such, the constraints must be solely expressed in terms of the variable of optimization, $\delta \mathbf{u}$. Output constraints imposed on equation (6.3), that satisfy the QP formulation, can be expressed as follows,

$$\hat{D}\delta \mathbf{u} \leq \bar{\mathbf{y}} - \mathbf{y}^0 - \hat{\mathbf{C}}\delta \mathbf{x}_k - \hat{\mathbf{E}}\delta \mathbf{d} \quad (6.20)$$

$$-\hat{D}\delta \mathbf{u} \leq \mathbf{y}^0 + \hat{\mathbf{C}}\delta \mathbf{x}_k + \hat{\mathbf{E}}\delta \mathbf{d} - \underline{\mathbf{y}} \quad (6.21)$$

where $\bar{\mathbf{y}} = [\bar{y}_k \ \bar{y}_{k+1} \ \cdots \ \bar{y}_{k+N}]^T$ is the vector of all output maximums over the prediction interval and $\underline{\mathbf{y}} = [\underline{y}_k \ \underline{y}_{k+1} \ \cdots \ \underline{y}_{k+N}]^T$ is the vector of all minimums. Finally, the control constraints in QP form are given as,

$$I_{N \times N}\delta \mathbf{u} \leq \bar{\mathbf{u}} - \mathbf{u}^0 \quad (6.22)$$

$$-I_{N \times N}\delta \mathbf{u} \leq \bar{\mathbf{u}} - \mathbf{u}^0 \quad (6.23)$$

where $\bar{\mathbf{u}} = [\bar{u}_k \ \bar{u}_{k+1} \ \cdots \ \bar{u}_{k+N}]^T$ is the maximum constraint on the input vector for time $k : k + N$ and $\underline{\mathbf{u}} = [\underline{u}_k \ \underline{u}_{k+1} \ \cdots \ \underline{u}_{k+N}]^T$.

The QP optimization routine detailed above must be solved at each time step k , to procure the input for the subsequent time step, δu_k^* . For clarity, the energy-aware MPC scheme is summarized.

1. Generate trajectories for linearization, \mathbf{x}^0 , with the desired command trajectory, \mathbf{u}^0 over the MPC horizon, N
2. Linearize equations (6.3) about the \mathbf{x}^0 trajectory yielding equations (6.5) and (6.6)

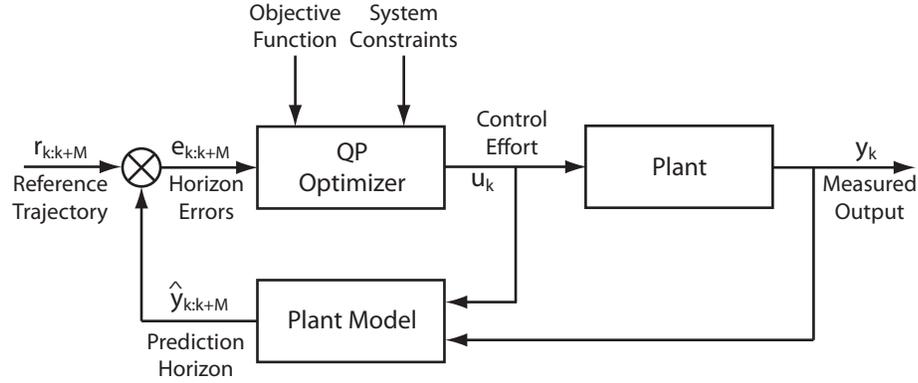


Figure 6.3: Block diagram representation of the receding horizon control scheme.

3. Solve the quadratic program for $\delta \mathbf{u}$,

$$\min_{\delta \mathbf{u}} J_k = \delta \mathbf{u}^T \left[\hat{B}^T \hat{Q} \hat{B} + \hat{R} \right] \delta \mathbf{u} + 2 \left[\hat{A} \delta x_k + \hat{E} \delta d \right]^T \hat{Q} \hat{B} \delta \mathbf{u} \quad (6.24)$$

$$\text{subject to } \hat{D} \delta \mathbf{u} \leq \bar{\mathbf{y}} - \mathbf{y}^0 - \hat{C} \delta \mathbf{x}_k - \hat{E} \delta d$$

$$-\hat{D} \delta \mathbf{u} \leq \mathbf{y}^0 + \hat{C} \delta \mathbf{x}_k + \hat{E} \delta d -$$

$$I_{N \times N} \delta \mathbf{u} \leq \bar{\mathbf{u}} - \mathbf{u}^0$$

$$-I_{N \times N} \delta \mathbf{u} \leq \bar{\mathbf{u}} - \mathbf{u}^0$$

4. Apply the optimal control, $\delta \mathbf{u}_k^*$ at time, k

6.2 Applications to Unmanned Ground Vehicles

The energy-aware MPC methodology presented above can be applied to any mobile system with finite onboard energy. In particular, the methodology excels where the onboard energy is stored in a rate-limited storage mechanism,

such as an electrochemical battery. Other energy storage devices, such as flywheels and ultra-capacitors, which do not exhibit significant rate-limits to energy, would not benefit significantly from the proposed energy-aware scheme. To collaborate with the other studies presented in this work, the energy-aware control methodology is applied on a simulated UGV system operating in a stochastic environment.

6.2.1 Longitudinal Ground Vehicle Model

To evaluate the performance of the proposed energy-aware MPC scheme, a simple longitudinal vehicle model was developed that integrates vehicle drivetrain dynamics with onboard battery dynamics through a power system. The model, schematically depicted in Figure 6.4, utilizes the same model described in Section 2.3.3 for battery dynamics. The vehicle/drivetrain dynamics discount slip and only provide for longitudinal motion with rolling resistance [75]. Conjoining the drivetrain and battery dynamics, the pulse-width modulation (PWM) power converter provides for control of the vehicle drive train via specification of the PWM duty cycle [0% - 100%]. As the dynamics of the battery and vehicle remain several orders of magnitude above the PWM model (measured greater than 1 kHz switching frequency for the DaNI UGV), a fast-average approximation for the power converter is adopted [146]. Fast averaging of the PWM dynamics results in a buck converter approximation which relates the input/output voltage, $V_m = V_b \times \delta(t)$, and current $i_b = i_l \times \delta(t)$.

Incorporating the PWM converter relationships into the vehicle/drivetrain

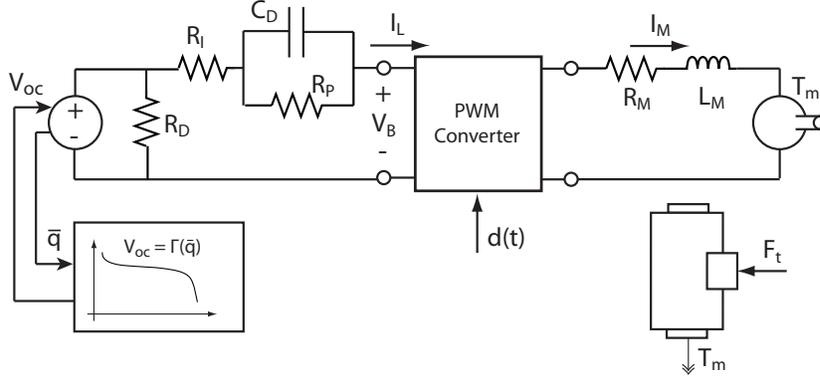


Figure 6.4: Longitudinal vehicle model for source-aware MPC algorithm assessment.

dynamics, the UGV velocity and motor currents can be found via the following expressions,

$$\dot{i}_m = -\frac{K_v}{r_w L_m} v_x - \frac{R_m}{L_m} i_m + \frac{1}{L_m} [\Gamma(\bar{q}) - V_D] u(t) - \frac{R_I}{L_m} i_m u^2(t) \quad (6.25)$$

$$\dot{v}_x = -\frac{b_w}{m_v r_w^2} v_x + \frac{K_t}{m_v r_w} i_m - g f_{(i)} \text{sgn} v_x \quad (6.26)$$

where v_x is the longitudinal UGV velocity, i_m is the drive motor current, $u(t)$ is the PWM duty cycle command, and $i_d(t)$ are additional battery current loads. Vehicle parameters include the vehicle mass, m_v , wheel radius, r_w , the drivetrain damping, b_w , and the extant terrain rolling resistance value, $f_{(i)}$. Additionally, the motor parameters are defined as the motor speed constant, K_v , the motor torque constant, K_t , the motor terminal resistance, R_m and the motor terminal inductance, L_m . Inclusion of the PWM algebraic relationships link the vehicle dynamics to the battery dynamics as the battery model states in equation (6.25) clearly illustrate. The battery dynamics, with the PWM

relationships, are given as follows,

$$\dot{V}_D = -\frac{1}{R_p C_D} V_D + \frac{1}{C_D} i_m u(t) + \frac{1}{C_D} i_d(t) \quad (6.27)$$

$$\dot{\bar{q}} = -\frac{1}{q_0 R_D} \Gamma(\bar{q}) + \frac{1}{q_0} i_m u(t) + \frac{1}{q_0} i_d(t) \quad (6.28)$$

where the parameters and states remain identical to those introduced in Section 2.3.3. Finally, the battery terminal voltage equations is given,

$$V_B = \Gamma(\bar{q}) - V_D - R_I i_m d(t) - R_I i_d(t) \quad (6.29)$$

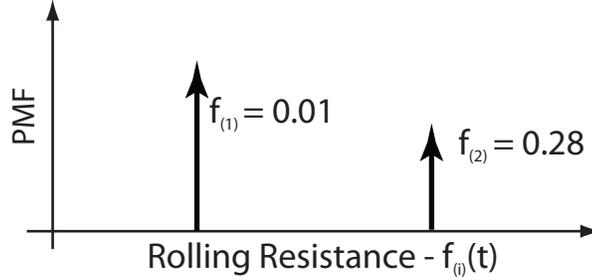


Figure 6.5: Rolling resistance stochastic load structure.

To simulate a field environment, a second-order Markov chain which represents the transition characteristics of the terrain rolling resistance was proposed. As the Markov process requires only two states, the nomenclature introduced in Section 4.3.1 is adopted. Rolling resistance coefficient adopted for this study were selected to approximate a smooth surface, $f_{(1)} = 0.01$, and rough terrain, $f_{(2)} = 0.28$ [75]. Furthermore the steady state characteristics were selected to ensure predominately smooth surface travel with transient rough patches, $\pi_1 = 0.7$ and $\pi_2 = 0.3$. A rolling resistance transition rate of 2.6 transitions per minute was selected via $\lambda = 0.95$.

6.2.2 Monte Carlo Simulation Study

In the following section, a Monte Carlo simulation study is presented in which the control fidelity of the proposed MPC algorithm is evaluated. As discussed, the proposed MPC algorithm is a command governing scheme that regulates input commands (as the PWM duty cycle) from a teleoperator/supervisory control algorithm contingent on the battery (or source) dynamics. To serve as a baseline, direct vehicle control (DVC) with no duty cycle restrictions is employed. The experimental studies conducted all utilized the DVC methodology where motor commands are directly implemented. For the stochastic switching environment proposed, the energy-aware control scheme was implemented to constrain the battery terminal voltage above a shutdown voltage of $V_{SD} = 9.5$ V. Simulation results of the MPC scheme preventing transient shutdown are illustrated in Figure 6.6.

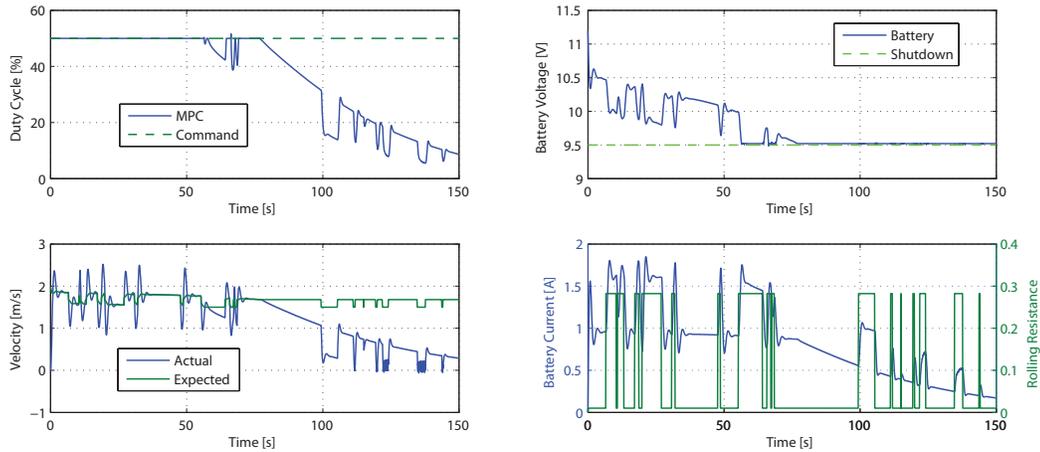


Figure 6.6: Energy-aware MPC simulation study illustrating transient shutdown prevention.

To contrast with the energy-aware MPC scheme, a constrained PID control methodology was also proposed as a heuristic solution to the energy-aware control problem. Contrasting with the MPC scheme which forecasts using the battery model, the PID methodology proposed uses only the battery output equation to solve for duty cycle values which prevent shutdown for the next time step, $k + 1$. For each time step, the PID control effort is computed via standard numerical PID operations, after which the resulting constraint is checked,

$$u_k \leq \frac{1}{R_I i_m(k)} [\Gamma(\bar{q}(k)) - V_{\min} - V_D(k) + R_I i_d(k)] \quad (6.30)$$

If the constraint is violated, a Newton-Raphson solver determines the critical duty cycle command, $u_k^{(c)}$, which results in shutdown. To ensure the voltage constraint is not exceeded, the constrained-PID command issued is $u_k^{(c)} - \varepsilon$, where ε is an *a priori* specified threshold. An example implementation of the heuristic constrained-PID method is illustrated in Figure 6.7.

The results for the energy-aware MPC and heuristic PID schemes in Figures 6.6 and 6.7 illustrate the methodologies over a single stochastic load realization. To analyze the potential benefits of the energy-aware schemes, 500 realizations of each of the three control methodologies (DVC,PID,MPC) were simulated until system shutdown. The UGV system was assumed to have shutdown if one of the three conditions was met:

1. Exceed cutoff voltage by $> 1\%$, where $V_{SD} = 9.5$ V

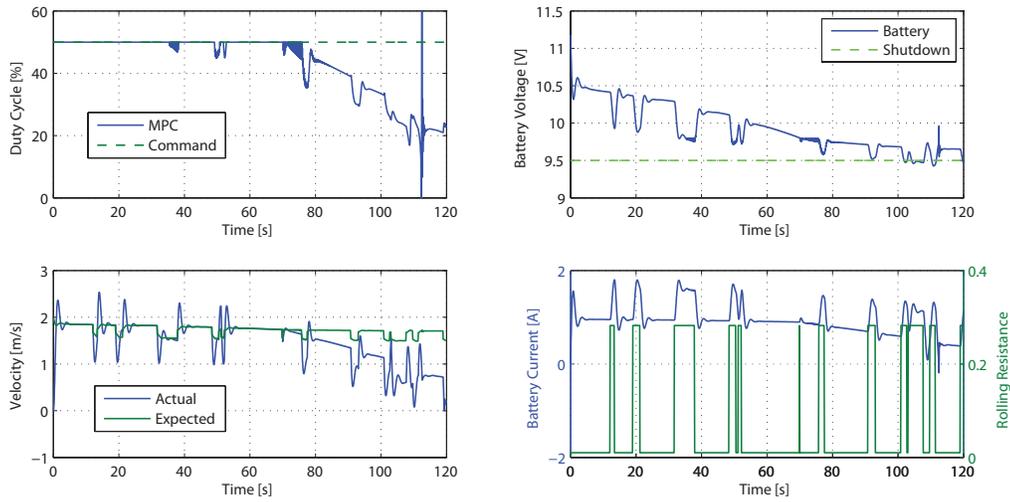


Figure 6.7: Heuristic constrained-PID for comparison to energy-aware MPC.

2. Ten second moving average of velocity drops below 25% of expected longitudinal rate
3. Quadratic program becomes infeasible for the MPC scheme

The resulting numerical PDFs for both the cumulative distance traveled and the total operational time for the simulations are illustrated in Figure 6.8. Furthermore, the failure modes of each controller type are summarized in Table 6.1. As expected, DVC failures resulted exclusively due to the battery terminal voltage exceeding the shutdown voltage. This follows as the control is not subject to any voltage constraint considerations. The PID scheme fails as a result of both exceeding the voltage and velocity thresholds. While the PID controller attempts to solve for a feasible control effort at each time step, the heuristic constrained PID does not predict future states. Thusly, the PID controller can implement a control action at time, k , resulting in any

control effort at $k + 1$ yielding system shutdown. Finally, the energy-aware MPC scheme was demonstrated to fail only as a result of plummeting vehicle velocity. Over the 500 realizations of the MPC scheme, the QP optimization was never infeasible.

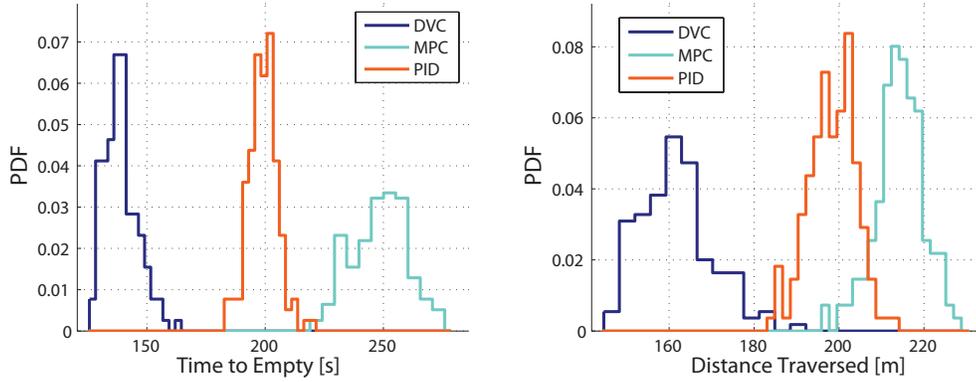


Figure 6.8: Monte Carlo comparison study results of energy-aware MPC as compared to DVC and PID. (a) Operational Time. (b) Cumulative distance.

Table 6.1: Energy-aware shutdown conditions for each presented control style.

Control	Battery Voltage	Low Velocity	QP Infeasible
DVC	100.0 %	0.0 %	-
PID	16.0 %	84.0 %	-
MPC	0.0 %	100 %	0.0 %

As can be seen in Figure 6.9, the energy-aware MPC algorithm outperforms both the heuristic constrained PID method and standard DVC in both the total operational time and the cumulative distance traveled. In terms of total operational time for all 500 simulations, the mean energy-aware MPC system operation time was 248.16 seconds. As such, the MPC scheme remained operational 48.96 seconds longer than the proposed heuristic PID method and

109.05 seconds longer than DVC. Furthermore, the mean cumulative distance traveled for the MPC scheme was 214.24 meters. Again, the MPC algorithm allowed the simulated UGV to travel an average of 15.86 meters more than the PID method and 52.61 additional meters over DVC.

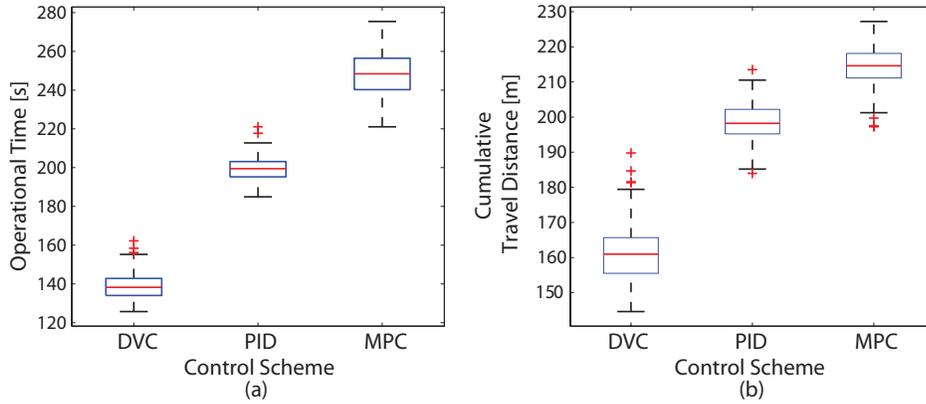


Figure 6.9: Box plot presentation of results of energy-aware MPC as compared to DVC and PID. (a) Operational Time. (b) Cumulative distance.

Finally, to determine whether computational requirements of the MPC algorithm outweighs the benefits of implementation, the computation time required for both the constrained PID and the energy-aware MPC were measured. Computation time included method initialization until an optimal δu_k^* is found. For the MPC algorithm, the mean computational time was found to be 6.06 ± 0.88 ms. Contrastingly, the mean computational time for the constrained PID algorithm was found to be 2.53 ± 1.78 ms. The box plot, in Figure 6.10, illustrates the quartile information of the computational time. As the proposed PID routine must solve a Newton-Raphson, the upper computational time bound depends on the required number of iterations for convergence. The

QP formulation of the energy-aware MPC algorithm with a single control variable, in contrast, is known to be exceedingly efficient in a computational sense [168].

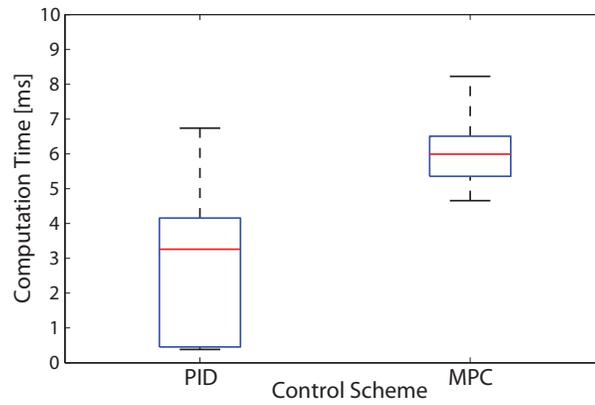


Figure 6.10: Comparison of computational times required over all UGV simulations.

Chapter 7

Conclusions

The goal of this work has been to extend online prediction/control capabilities for mobile systems with finite onboard energy that operate in unstructured environments. These mobile systems, such as ground robots and electric vehicles, have been shown to experience significant transient power demands that invalidate the assumptions for existing battery remaining runtime and energy-aware control schemes. As such, the relevant literature for run-time prediction, probability of mission completion prediction and energy-aware control methodologies was reviewed. The GMJM/PF integrated prediction scheme was proposed/developed such that existing prediction limitations could be overcome, and experimentally demonstrated via two distinct mobile system discharge studies, i.e. the Packbot and DaNI UGV. A reliability-based PoMC assessment algorithm was introduced to include battery dynamics into mission probability of completion prediction, and the methodology was demonstrated with experimental studies involving the DaNI UGV. Finally, an online energy-aware MPC algorithm was developed to mitigate current transient related shutdowns frequently encountered in field operations.

7.1 Summary and Directions for Future Work

In the following section, the research presented in this dissertation is recapitulated. Furthermore, the conclusions drawn from algorithm development and experimental implementation are restated. Finally, any avenues for potential future work are presented and discussed.

7.1.1 Online Unsupervised Transient Load Characterization

The novel online GMJM load characterization scheme was proposed to address the need for online characterization of *a priori* unknown transient load structures encountered by mobile systems operating in unstructured environments. As discussed battery systems remain prone to transient shutdown conditions and the rate-capacity effect. As a result, for battery run-time prediction, the transient characteristics of the battery loading process must be preserved. The GMJM method, introduced in Chapter 3, adopts and integrates existing mathematical methods to provide for an online scheme for characterization of stochastic transient loads for load forecasting.

As the loading process for a mobile system is *a priori* unknown, the transient characterization scheme must be capable of unsupervised load clustering. The Gaussian mixture clustering method was adopted with the Akaike information criterion metric to ascertain the optimal number of loading clusters and cluster statistics. As cluster number identification occurs online, the developed methodology remains self-supervised. After load cluster identification, the transient characteristics of the loading process are fit to a jump-

Markov process via maximum likelihood parameter estimation. The integrated Gaussian mixtures and jump-Markov process form a fully characterized transient load process that can be used for load forecasting and battery run-time prediction. The GMJM load characterization algorithm was experimentally implemented and demonstrated with two UGV studies.

To improve on the GMJM transient load characterization algorithm, the stochastic process characterization of each loading cluster could be further modeled. In regions of identical clusters, a time series process, such as the autoregressive moving average, can extend the model to include the spectral characteristics of the power loads. Improvement of prediction fidelity is conjectured to remain minimal as a result of this improvement, as the primary load considerations are the transient jumps. An additional improvement to the GMJM scheme is the use of an apparent relationship between cluster separation, cluster transition rates, and the necessary data horizon for load characterization. An analytical relationship between these variables would allow for proper load horizon selection without residual evaluation for an adaptive horizon or *a priori* horizon selection based on expected loads.

7.1.2 Battery Remaining Run-time Prediction via the Particle Filter

To overcome the existing limitations of battery remaining run-time prediction methodologies, the model-based particle filter has begun to see adoption in the literature. However, existing implementations of the particle filter

for battery remaining run-time prediction do not account for the transient characteristics of the loading process encountered by mobile systems operating in unstructured environments. As a result, unless the future transient loads are *a priori* known, the particle filter predictor gains little over the EKF predictor. As a result, the proposed/implemented methodology, introduced in Chapter 4, combines the predictive power of the PF with the GMJM transient load structure.

To predict battery remaining run-time, the PF represents all state uncertainty as the sum of weighted particles which propagate through the non-linear dynamic battery model. Each state particle is acted on by an individual realization of the GMJM load. Resultantly, the battery RRT probability distribution can be predicted. Furthermore, the GMJM/PF scheme was demonstrated to excel over existing RRT prediction methods when load transients exceed 20W. As implementation of the GMJM/PF methodology requires two orders of magnitude more computational power than linear model-based techniques, the assessment of expected load transients provides for design considerations for selecting a battery RRT scheme. Finally, the GMJM/PF method was demonstrated to outperform all existing methods via two experimental UGV studies.

In terms of potential future work, the GMJM/PF battery run-time predictor computational efficiency could be further improved. As discussed, the algorithm developed in this work required two orders of magnitude more computational time than a linear variance (EKF) predictor. Improvement in

computational efficiency could result from implementation of next generation predictors, such as the Gaussian-sum filter or weighted propagation of Gaussian mixtures through nonlinear systems. These non-Gaussian approaches could offer a hybrid prediction scheme where a Gaussian-sum filter is used for initial prediction until a maximum current results in system shutdown. After which the PF scheme could be initialized for transient shutdown prediction. Additional improvements in prediction fidelity could be achieved with more accurate battery models for both online state estimation and model-based prediction. However, these gains would come at the expense of additional computational complexity.

7.1.3 Online Mission Reliability Assessment

A probability of mission completion algorithm was introduced in Chapter 5 for online assessment of mission reliability. Most reliability analysis methods, for mobile systems, consider mobility or component failure conditions. Furthermore, posited methods, with energy constraints, fail to address the dependency of available onboard energy on mission characteristics. To overcome existing limitations, a method was presented for online characterization of a bivariate mission process relating overall mission time and battery run-times. Prediction of the probability of mission completion could be found directly by numerical quadrature of the PDF over the failure region where mission time exceeds available battery run-time.

For online PoMC prediction, the mobile system mission process was

modeled as a bivariate normal distribution. Battery RRT predictions via the GMJM/PF and mission time predictions via EWMA forecasts of velocities provide a prediction pair, (OBRT, OMT). With these prediction pairs, the correlation is estimated via Bayesian inferencing. With the marginal distributions of battery run-time/mission time and the correlation estimate, a bivariate formulation of the mission process is ascertained. Integration of the bivariate distribution over the region where the mission time exceeds the battery run-time yields the PoMC estimate. The PoMC estimator was implemented with experimental UGV data and demonstrated to accurately predict the ground truth mission probabilities for a range of mission profiles. Finally, the importance of mission time and battery time correlation was demonstrated via an experimental comparison against the assumption of independence.

In terms of future work, the PoMC estimation algorithm could be improved by incorporating the non-Gaussian RRT prediction characteristics into the mission process characterization. Using the theory of copulas, non-Gaussian marginal distributions can be used to generate a multivariate non-Gaussian distribution function [143]. As such, prediction inaccuracies resulting from the bivariate approximation can be mitigated. Copula theory, however, requires numerical computation to implement and is expected to require additional computational power over the bivariate assumption. Another avenue for algorithm improvement is the potential for task time updating online. Provided repeated tasks and online measurements of task times, a Bayesian updating scheme could be used to update task times during mobile system

operation. Finally, the additional forecasting methodologies for the prediction of overall mission run-time could be implemented and contrasted with the selected EWMA scheme.

7.1.4 Transient Shutdown Prevention Control

Finally, energy-aware control was extended to include mobile systems that operate with stochastic transient loads in Chapter 6. Model predictive control, in the literature, has been utilized for mobile system control and energy allocation, but only in the case of mobile systems with multiple energy sources. Contrastingly, the proposed energy-aware MPC scheme was designed to prevent transient shutdown conditions of the *sole* onboard battery system. Furthermore the proposed MPC scheme was developed to incorporate characterized stochastic loads, such as the transient loads characterized by the GMJM process.

The energy-aware MPC scheme was developed to take a command from either a vehicle teleoperator or supervisory control system as an input. The nonlinear battery/vehicle model is then linearized about the specified command trajectory and the system model is formulated as a linear time-varying perturbation model. The model predictive control scheme is formulated as a quadratic program that is then efficiently solved via an active set solver routine. The MPC routine solves for the optimal control command input that tracks the nonlinear trajectory subject to the battery shutdown voltage constraint. Monte Carlo simulation realizations of the MPC scheme were shown

to outperform direct vehicle drive and a heuristical constrained PID control scheme.

As the presented energy-aware model predictive control scheme is only preliminary, ample future development opportunities remain. At the writing of this dissertation, the energy-aware algorithm was in the process of being implemented on the DaNI UGV. Online implementation requires battery state-of-charge estimation for the battery model states and disturbance estimation as present disturbances were assumed to be known. Additional future work involves extending the MPC routine to include robust MPC methods to account for load uncertainties. Recent advances in tube-based robust model predictive control serve as an auspicious method extension.

7.2 Significant Scientific Contributions

Several contributions have been made to the fields of battery remaining run-time prediction, mobile system (in particular UGVs) mission reliability analysis, experimental analysis of stochastic ground robotic processes, and energy-aware control. The comprehensive list of contributions is presented as follows:

1. **Self-supervised Gaussian mixture jump-Markov load characterization algorithm:** A method for online transient load characterization, presented in Chapter 3 was developed. The GMJM algorithm integrates Gaussian mixture clustering with the Akaike information cri-

terion for self-supervised identification of the number of transient load structures. Furthermore, the load transition rates are characterized via a jump-Markov process. While the individual methods remain well-known, the novel combination and application serve as a contribution of this work.

2. **Battery remaining run-time prognostics via the particle filter with transient load forecasts:** A method for battery remaining run-time prediction, introduced in Chapter 4, was developed which integrates the model-based particle filter prediction routine with the GMJM load process. Traditional battery run-time prediction has neglected transient loads which disregard the dynamic nature of battery systems. The GMJM/PF scheme was also extended to include uncertain battery shutdown conditions via the reliability integral.
3. **Online probability of mission completion assessment for mobile systems:** A method for predicting the probability of completing a mission was presented in Chapter 5. Proposed PoMC techniques have discounted the dependence of the battery system on the operational characteristics of the mission. The method presented in this work overcomes this limitation by performing Bayesian correlation estimation of the mission process. Standard reliability methods are used to compute the probability of the overall mission time exceeding the overall battery run-time.
4. **Energy-aware model predictive control for transient shutdown**

prevention: A method for transient shutdown prevention was proposed which utilizes receding horizon model predictive control. Existing applications of energy-aware control either control distribution of power between energy sources (onboard generator and batteries) or prevent battery shutdown on systems via task scheduling, such as wireless networks. The introduced method in Chapter 6 extends the energy-aware MPC methodology to mobile systems with stochastic loads through an integrated battery/vehicle system. The quadratic programming formulation of MPC was adopted for real-time control.

5. **Experimental stochastic terrain environment for UGV discharge**

studies: An indoor experimental test stand was designed and constructed for repeated UGV discharge tests. In the literature, existing run-time prediction schemes largely assess prediction fidelity with a single discharge experiment. The developed terrain environment provided for repeated UGV discharge studies which simulated the unstructured power demands experienced during UGV field operation. All methodologies presented in the work were statistically demonstrated over numerous experimental studies.

Appendices

Appendix A

Modeling, Estimation and Parameter Identification of Dynamic Battery Systems

The following appendix details the experimental work conducted for the parameter identification of the 12V 3000 mAh NiMH battery packs used for the experimental studies in this work. Furthermore, the experimental work conducted to characterize the battery protective circuitry used in the NI DaNI ground robot is presented. Finally, while not explicitly discussed in the body of the dissertation, the extended Kalman filter for battery SOC estimation is presented. Online battery RRT prediction require either direct measurement or estimation of battery SOC information for an accurate model-based prediction.

A.1 Parameter Identification and Optimization

Lumped parameter models were presented for a generic battery system throughout this work. Application of the ECM for estimation, control and prediction requires the population of model parameters and identification of the nonlinear discharge curve, given by equation (2.21). Electrochemical impedance spectroscopy (EIS) has been demonstrated in the literature as an

accurate technique for characterization of the lumped resistances and capacitance of the ECM in Figure 2.13, resulting in model errors of less than $\pm 0.2\%$ [169]. However, impedance measurements require signal magnitude and phase information, typically acquired via a lock-in amplifier [170] or galvanostatic spectroscopy [171], both which were beyond the budget/scope of this research.

Consequently, the parameters were optimized via nonlinear least squares to minimize the residues of the model voltage and measured voltage [170]. Periodic discharge tests were conducted on battery packs, for load current and measure voltage information, via a constructed current controlled discharge test stand. In section A.1.1, the experimental discharge setup for parameter identification is presented, and the experimentally ascertained open circuit discharge curve of a 3000 mA-h NiMH battery pack is shown. The nonlinear least squares transient parameter optimization routine is briefly discussed in section A.1.2, followed by the optimized values.

A.1.1 Battery Discharge Experiments

Prior to the dynamic optimization of ECM parameters, discharge curve must be identified. Since the discharge curve, equation (2.21), relates the SOC to the open circuit voltage, the voltage measurements require a recovered unloaded battery for each particular charge [7]. As a result, loading the battery periodically decreases the SOC during the loading cycle, and the unloaded duration provides for ion diffusion recovery to the new open circuit voltage. A current controlled discharge test stand, shown schematically in Figure A.1(a)

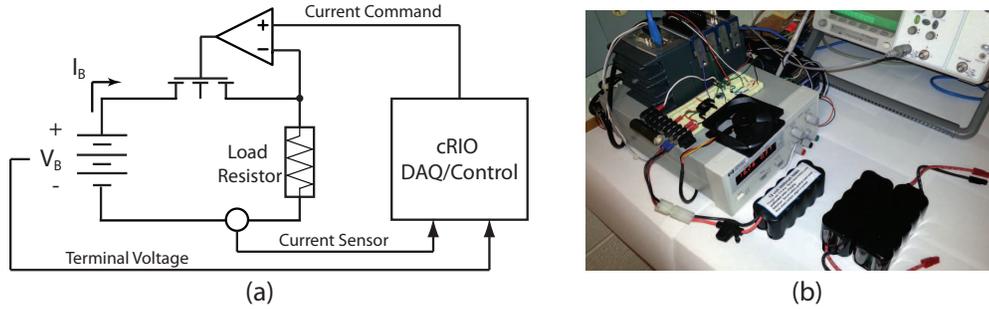


Figure A.1: (a) Schematic of the current controlled discharge setup. (b) Laboratory realization of the discharge setup.

and physically in Figure A.1(b), was constructed for controlled periodic discharge.

A 3000 mA-h NiMH battery pack, used in all future experiments, was subjected to a 1/20 Hz periodic discharge at 1 A. The discharge period was selected such that the measured voltage would return within 1% of the expected open circuit voltage during the recovery phase, seen in Figure A.2(a). A high discharge rate was desirable to minimize test time; however, hardware limitations required discharge currents under 3A. The open circuit voltage values were found by extracting the measured terminal voltage prior to the next discharge cycle.

A bounded general polynomial fit, given by equation (A.1), provides a mathematical depiction of the nonlinear discharge curve,

$$\Gamma(\bar{q}) \approx \sum_{n=1}^N a_n x^n \quad (\text{A.1})$$

where N is the polynomial order and a_n are polynomial coefficients. Utilizing the open circuit voltage data from two full discharge tests, a general

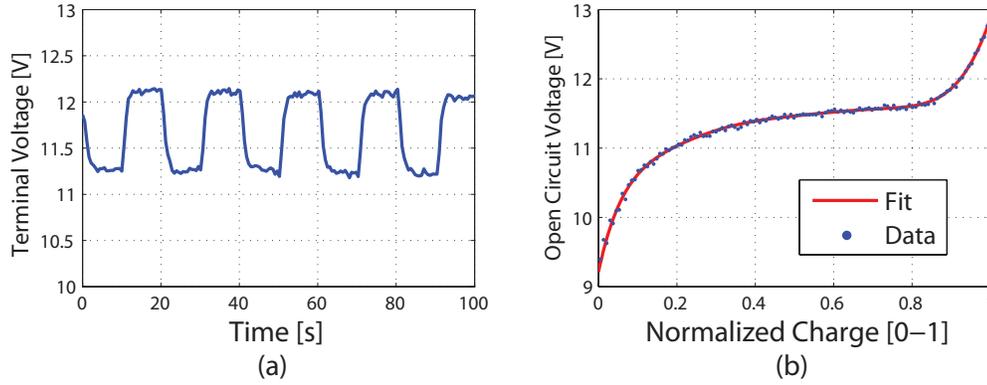


Figure A.2: (a) Selection of discharge data illustrating the recovery effect returning to the open circuit voltage. (b) Polynomial fit of the open circuit voltage data.

Table A.1: Discharge curve polynomial fit.

Parameter	Value	Parameter	Value
a_0	9.246	a_4	-1.121×10^3
a_1	23.856	a_5	1.393×10^3
a_2	-143.77	a_6	-930.83
a_3	521.97	a_7	259.60

polynomial least-squares algorithm was employed to determine the leading coefficients, $a_n a_0$ [76]. The polynomial order was determined via a comparison of normalized residuals [85]. Beyond the eight degree polynomial, the decrease in the magnitude of the residual remained minimal. Consequently, the discharge curve is represented as an eight order polynomial with the following terms:

A.1.2 Parameter Optimization Routine

To identify the ECM lumped parameters, a constrained nonlinear least squares optimization routine was implemented to minimize residuals between the model prediction and experimental measurements, akin to [172]. Succinctly stated, the optimization problem takes the following form,

$$\min_{\theta} . \quad (\hat{V}_B(\theta) - V_B^{(obs)})^T (\hat{V}_B(\theta) - V_B^{(obs)}) \quad (\text{A.2})$$

$$\text{subject to : } \dot{V}_D = f_1(V_D, \bar{q}, I_L, \theta) \quad (\text{A.3})$$

$$\dot{\bar{q}} = f_2(V_D, \bar{q}, I_L, \theta)$$

$$\theta \in \Theta$$

where \hat{V}_B is the predicted terminal voltage by the model, $V_B^{(obs)}$ is the measured terminal voltage during an experiment, θ is the set of unknown parameters to be optimized, and Θ is the feasible set for the model parameters. The parameter vector, θ , contains the set of system parameters, $\theta = \{R_P, C_D, R_I, R_D, q_0\}$, which serves as the variable set of optimization. The dynamic constraint equations, $\dot{V}_D = f_1(V_D, \bar{q}, I_L, \theta)$ and $\dot{\bar{q}} = f_2(V_D, \bar{q}, I_L, \theta)$, enforce the Thévenin ECM dynamic equations (2.22) and (2.23), respectively.

Excitation of the dynamic states, V_D and \hat{q} , was achieved via multiple discharge tests with large transient loads [170]. Periodic discharge currents of 0-3A at a rate of 0.25 Hz and 0.5 Hz captured significant transients for optimization analysis, as compared to the open circuit discharge curve discharge experiments. Parameter optimization performed at varied SOCs and at differing discharge rates potentially avoids non-optimal parameter identification,

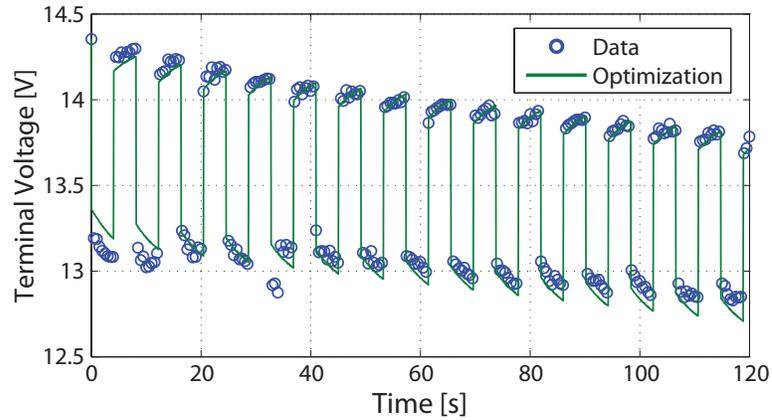


Figure A.3: Comparison of simulated optimal parameters and discharge voltage data of a NiMH battery pack.

due to the nonlinearity (and local minima) of the optimization problem (A.2) [172]. Furthermore, the set of parameters to be optimized are initialed with parameters given in [173] for a NiMH battery pack. Parameters were bounded by physically imposed constraints, e.g., no negative resistances or capacitances.

Using a generic interior point optimization routine, the optimization problem, (A.2), was solved with multiple initializations. A particular parameter identification optimization is illustrated in Figure A.3, where a simulation of the Thévenin with optimal parameters is compared with the measured voltage data. Averaging the optimized system parameters provides the values presented in Table A.2.

Table A.2: Optimized parameters for a 3000 mA-h NiMH battery pack.

Parameter	Value	Description
C_D	$73.7F$	Diffusion Capacitance
R_D	$2.59 \times 10^3\Omega$	Dissipation Resistance
R_P	0.068Ω	Polarization Resistance
R_I	0.327Ω	Internal Resistance
q_0	9.74×10^3C	Initial Charge

A.2 Identification of Shutdown Conditions

During field operation, battery shutdown does not necessarily commence upon full depletion of the internal charge [62]. A drained charge remains intrinsically linked to the open circuit voltage of the battery cell and the drained voltage is arbitrarily linked, typically by the manufacturer, to cell chemistry to preserve battery endurance [7]. To enforce consistent shutdown of the powered system and prevent cell damage resulting from deep discharge (particularly for Li-ion cells), modern switched-mode power electronic converters and battery protective circuitry supplement the battery system [174]. Resultantly, the shutdown of a mobile system powered via a battery depends on the shutdown conditions of the protective circuitry, which ultimately depend on the measured battery terminal voltage. While remaining battery SOC contributes to terminal voltage, Figure (voltage polarization) and equation (2.24) illustrate other contributing factors, such as the diffusion voltage and ohmic voltage drop due to the load current. Consequently, run time prediction must account for voltage-based shutdown conditions, and hence these conditions must be determined.

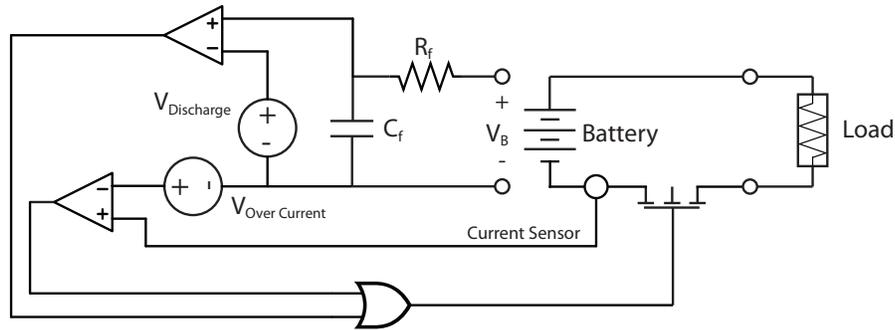


Figure A.4: Battery Protective Circuitry which enforces voltage shutoff and over-current protection.

Fundamentally, a battery protective circuit prevents damage to the battery cells via shutdown in the event of two primary scenarios: over current draw and low terminal voltage [174]. A simplified protective circuit, illustrated in Figure A.4, terminates system operation in the event of a detected over current or low battery terminal voltage. More complex protective system exist which utilize additional temperature measurements for shutdown, in addition to state-of-health monitoring and cell rebalancing [174]. For the experimental work in this dissertation, a DC-DC boost converter (XP JCL30 series) was used for battery protection which has a surge current shutdown condition of $4.7A$. The maximum current encountered during experiments was $3.45A$ and as a result, we focus on voltage shutdown conditions characterization for this work.

Commonly, protective circuitry does not provide information on shutdown conditions, and resultantly, the statistics of voltage shutdown must be ascertained experimentally, with the methodology illustrated in Figure A.6.

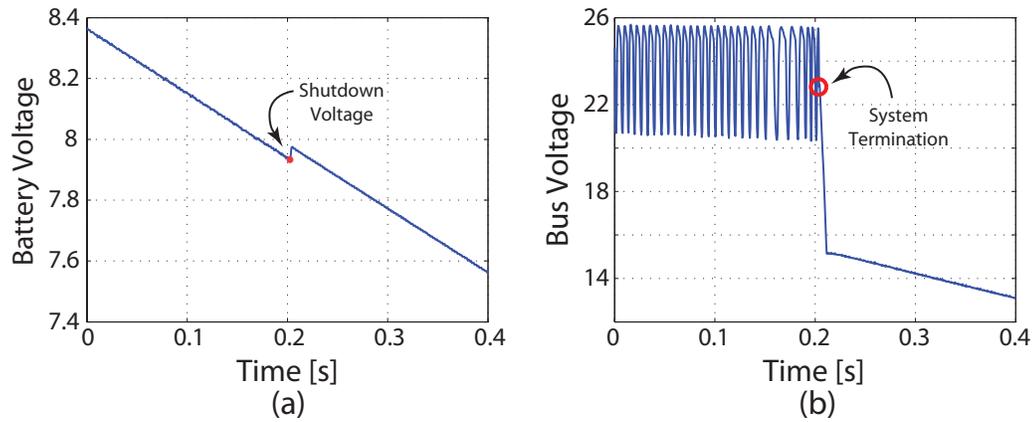


Figure A.5: (a) Ramped voltage decay of “battery” system. (b) Transient drop of the DC-DC boost converter output.

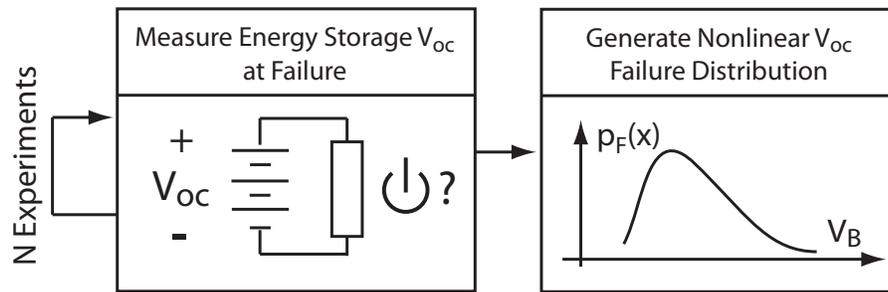


Figure A.6: Methodology for evaluation of shutdown characteristics of a battery power converter/protective circuit.

Substitution of a ramped voltage supply for the battery provides for a repeatable method for shutdown voltage measurement. Voltage supplied to the DC-DC boost converter is decreased via a ramp function until the output of the converter drops below 5% of the regulated output voltage (in our case, a 24V line bus). Sample experimental data is illustrated in Figure A.5.

The experiments, illustrated in Figure A.6, were repeated until the first four statistical moments (mean, variance, skewness, and kurtosis) converged.

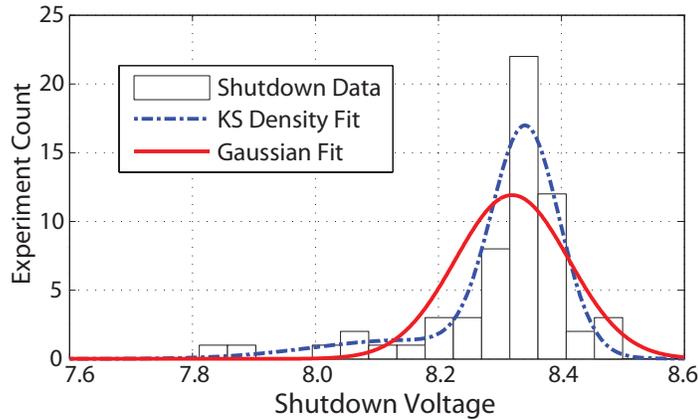


Figure A.7: Experimentally measured shutdown voltage probability density function for the XP JCL30 protective circuit.

For the XP JCL30 DC-DC boost converter, thirty experimental iterations were required. Data and a kernel density estimation fit are illustrated in Figure A.7 with an additional Gaussian fit. Nominally, the system will shut down when the battery terminal voltage descends below 8.22 ± 0.277 V. Due to the minimal variance associated with shutdown conditions, battery shut down voltage will be a scalar value of $8.22V$ unless otherwise noted.

A.3 Online State-of-Charge Estimation

Remaining run time algorithms, as introduced in Section 1.2.1, require either direct knowledge/ real-time measurements or some form of online estimation of battery state-of-charge. As illustrated in Section 2.3.3 in equation (2.21), the battery open circuit voltage depends on the battery SOC through nonlinear discharge curve homeomorphism. Resultantly, the battery terminal voltage depends primarily on the remaining charge in the battery cells [95].

Given the dependency of the battery protective circuitry and power systems, discussed in Section A.2, on the instantaneous terminal voltage of the battery, knowledge of battery SOC remains paramount for the prediction of run time and shutdown conditions.

A diverse spectrum of methodologies exist in the literature for SOC estimation which vary significantly with regards to the hardware implementation, algorithm and computational complexity and eventual estimation accuracy. A review of techniques for implementable SOC estimation for devices is presented in [175]. Direct quantization approaches, such as the physical measurement of cell electrolyte specific gravity and impedance spectroscopy, provide accurate estimations of the remaining charge (within 1%) [95]. However, each of these methodologies requires temporary halting of current draw from the cell for measurement, and resultantly, neither technique could be implemented for real-time SOC estimation on a mobile system [169].

Contrastingly, system estimation algorithms, which suffer in accuracy over physical measurements, have supplanted other direct techniques for real-time SOC estimation on mobile devices with minimal computational power. State estimators, such as the extended/unscented Kalman filter, require only basic matrix manipulation computational methods and digital measurements, and thusly, can be implemented directly on a mobile system [48]. Additionally, provided an appropriate fidelity battery model, online estimation schemes commonly report SOC estimation errors of less than $\pm 5\%$ [95]. Furthermore, implementation successes for model-based SOC estimation have been reported

with a variety of applications ranging from mobile hybrid electric vehicles to stationary photovoltaic non-dispatchable energy storage/ generation [175].

As was illustrated in Section 2.3.3, the battery ECM utilized for prediction contains nonlinear components. Resultantly, the model-based estimator must be capable of nonlinear state estimation. Fortunately for the sake of brevity, the techniques discussed in the literature for nonlinear state estimation for battery SOC remain identical to the prediction methodologies discussed in Section 2.1. State estimation, as contrasting with state prediction, requires an additional update step which utilizes information from a current measurement to update the prediction from a dynamic model. While the unscented Kalman filter [176], hybrid estimator [177], or the particle filter [55] have been shown to be efficacious in battery SOC estimation, computationally light extended Kalman filter demonstrates accurate estimations with errors under 10% [175]. Additionally, the Kalman filter methodology includes estimates of the covariance matrix which can be incorporated in the state uncertainty for prediction. A brief review of state estimation and the EKF is presented, followed by a discussion of the implementation of the EKF into the PF prediction methodology of Chapter 4.

Anecdotally, the Kalman filter can be interpreted as a method of “fusing” information from both real-world sensor information and first-principles based dynamic model information. Properties of a dynamic systems, which remain tedious or impractical to measure directly, can be estimated with information from both measurements and a dynamic model. In the case of the

battery system, direct measurement of SOC remains unviable in mobile systems and thusly state estimation has been applied for SOC estimation with success [44].

In the theoretic sense, the estimator takes information from measurements and the output of the dynamic model and attempts to “drive” the innovation term (residual between the model and reality) to zero, see Figure A.8. The estimator accomplishes this convergence to zero innovation by sequentially updating the state estimates given new measurement information. Furthermore, the Kalman gain (which determines the rate of change of states) is determined by the relative “trust” of information from measurement and the model respectively. This “trust” is represented by the covariance (or uncertainty) associated with the states, measurement noise and model (process) noise. For a more in depth coverage of nonlinear estimation theory, please consult [48].

State estimation of battery SOC requires a model of the system dynamics. Fortunately, the equivalent circuit model given in Section 2.3.3 has been shown accurately estimation battery SOC within a $\pm 5\%$ window [175]. Additionally, state estimation remains feasible given satisfaction of the observability condition [167]. The proof of observability of equations (2.22) and (2.24) is provided in Section A.4. For the extend Kalman filter, the nonlinear state space equations are sequentially linearized about the state trajectory, such that the customary Kalman theory equations become applicable [48]. Linearization of the continuous nonlinear equations given by equation, yield

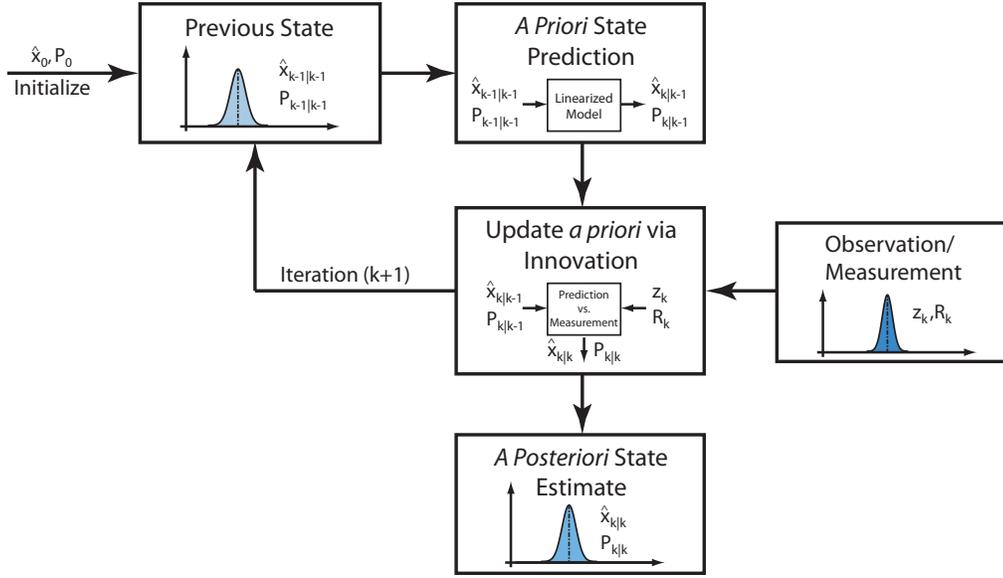


Figure A.8: Overview of extended Kalman filter estimation scheme.

the following state matrices,

$$A = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}(t)} = \begin{bmatrix} -1/R_P C_D & 0 \\ 0 & -\Gamma_{(1)}(\bar{q}_0)/q_0 R_D \end{bmatrix} \quad (\text{A.4})$$

$$B = \left. \frac{\partial g}{\partial x} \right|_{\hat{x}(t)} = \begin{bmatrix} 1/C_D \\ -1/q_0 \end{bmatrix} \quad (\text{A.5})$$

where $\Gamma_{(1)}(\bar{q}_0) = \partial\Gamma(\bar{q}_0)/\partial\bar{q}$ which is the first derivative of the continuous discharge curve function, which by definition is a C^∞ function on the interval $\bar{q} \in [0, 1]$. For extended Kalman filtering, the model output equation, see equation (2.24), can still remain nonlinear since the output information is only necessary for the innovation calculation. Resultantly, the innovation for the SOC estimator is as follows,

$$e_k = V_B - \Gamma(\hat{q}) + \hat{V}_D + R_I I_L \quad (\text{A.6})$$

where e_k is the estimator error or innovation, V_B is the measured battery terminal voltage, \hat{q} is the estimated state of charge, \hat{V}_D is the estimated diffusion voltage and I_L is the measured current load. Finally, due to the relative simplicity of the linear state equations which result from linearization, further discretization of the matrices provides for a computationally efficient estimator [24]. Discretization of linear state space matrices remains a straightforward analysis of the matrix exponential as follows [48],

$$A_D = e^{A\Delta T} = \begin{bmatrix} e^{\Delta T/C_D R_P} & 0 \\ 0 & e^{-\Gamma_{(1)}(\bar{q}_0)\Delta T/q_0 R_D} \end{bmatrix} \quad (\text{A.7})$$

$$B_D = \left[\int_0^{\Delta T} e^{A\sigma} d\sigma \right] B = \begin{bmatrix} -R_P (e^{-\Delta T/C_D R_P} - 1) \\ \frac{R_D}{\Gamma_{(1)}(\bar{q}_0)} (e^{-\Gamma_{(1)}(\bar{q}_0)\Delta T/q_0 R_D} - 1) \end{bmatrix} \quad (\text{A.8})$$

where ΔT is the discrete time step. With the above discrete time matrices, the extended Kalman filter can be formulated without continuous dynamics. Resultantly, the *a priori* prediction equations for the state estimates and the state covariance matrix can be expressed as follows,

$$\hat{x}_{k|k-1} = A_D \hat{x}_{k-1|k-1} + B_D u_{k-1} \quad (\text{A.9})$$

$$P_{k|k-1} = A_D P_{k-1|k-1} A_D^T + Q_{k-1} \quad (\text{A.10})$$

where Q_k is the process noise covariance which adds directly to the state covariance matrix, and R_k is the measurement noise variance which represents uncertainty associated with the measurement process. Furthermore, $\hat{x}_{k-1|k-1} = [\hat{V}_D(k-1|k-1) \quad \hat{q}(k-1|k-1)]^T$ is the previous state estimate and $\hat{x}_{k|k-1}$ is the *a priori* state estimate (prediction step only). Finally, u_k is

the current load on the battery, I_L , and $P_{k|k-1}$ is the covariance matrix. Given the measurement, the *a priori* predictions can be updated with the innovation term in equation (A.6). The Kalman gain, which optimally updates the state estimates to minimize the mean-square error of the states, is given as follows,

$$K_k = P_{k|k-1} B_D^T [B_D P_{k|k-1} B_D^T + R_k]^{-1} \quad (\text{A.11})$$

Provided both the Kalman gain of equation (A.11) and the *a priori* estimates of equations (A.9) and (A.10), the computation of the *a posteriori* state estimates and covariance matrix remains straightforward as follows,

$$\hat{x}_{k|k} = \hat{x}_{k-1|k-1} + K_k e_k \quad (\text{A.12})$$

$$P_{k|k} = [I - K_k B_D] P_{k|k-1} \quad (\text{A.13})$$

where $\hat{x}_{k|k}$ is the updated state estimate and $P_{k|k}$ is the updated covariance matrix, both of which are used for the prediction methodology in Section 4.1.1. Additionally, e_k comes from the measurement, shown in equation (A.6).

To confirm the efficacy of the SOC estimation algorithm discussed above, the methodology was implemented with the ten cell 3000 mA-h NiMH battery pack which was used for parameter identification, in Section A.1. Resultantly, the parameters and discharge curve, $\Gamma(q)$, used in the EKF remain identical. A stochastic discharge current, shown in Figure A.9(b), ranging from 0-3 amps was drawn from the battery pack via robotic vehicle which is discussed in Section 4.4.2. This uncertain current produced a battery terminal voltage discharge curve shown in Figure A.9(a). Implementation of the filter

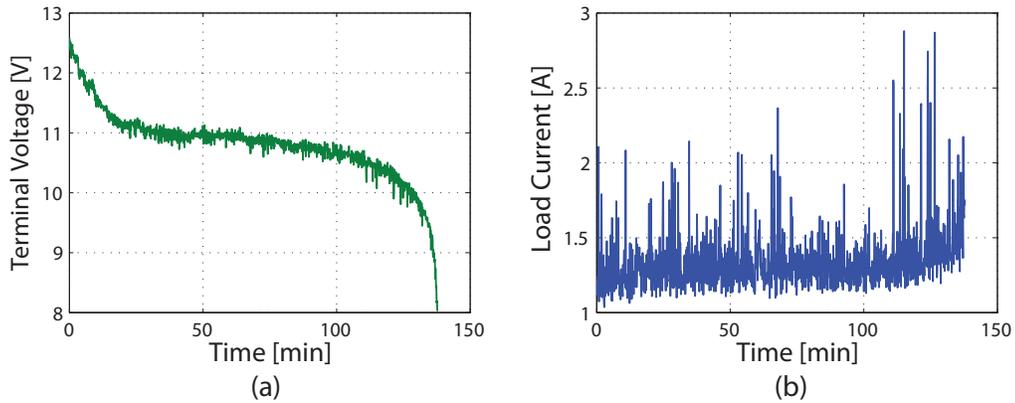


Figure A.9: (a) Battery discharge curve. (b) Stochastic current load applied to the NiMH battery pack for 2.25 hours.

theory discussed above provides the SOC estimation for the entire time interval of testing. The resulting state of charge mean estimate with the variance confidence interval are shown in Figure A.10. As previously discussed, the variance uncertainty associated with the SOC estimate can be directly utilized in the PF prediction algorithm discussed in Section 4.1. Preservation of the variance associated with estimation for prediction ensures proper representation of all uncertainty through all algorithms, as illustrated in Figure A.8. Resultantly, the initial uncertainty used to generate particles for prediction comes from $P_{\bar{q}}$ which is illustrated in Figure A.11.

A.4 Nonlinear Observability of Battery Model

The feasibility of state estimation theory, discussed in Section A.3, is contingent on the observability of the dynamic system. While the general theory of observability relies on several nuanced technical definitions, colloquially,

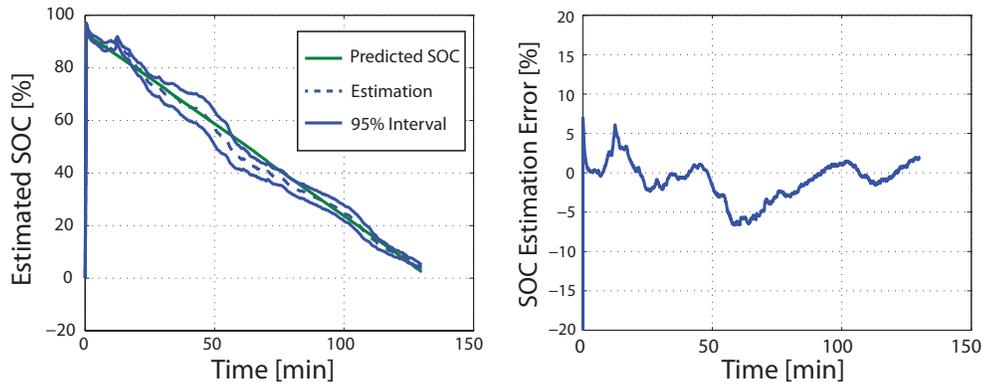


Figure A.10: State of charge estimation via the extended Kalman Filter and estimator error.

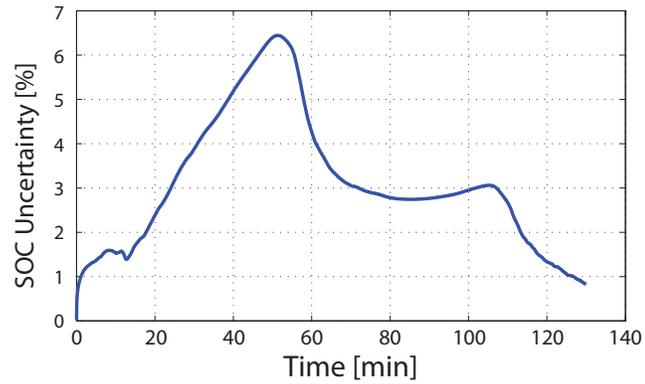


Figure A.11: State of charge variance used for initialization of the particle filter algorithm.

observability of a system implies that the internal model states can be uniquely determined through the dynamics given a finite number of measurements.

Definition A.4.1. Observability for a continuous system ensues if and only if for some arbitrary initial state $x(0)$, the value of $x(0)$ can be ascertained uniquely via only the inputs, $u(\sigma)$, and the outputs, $y(\sigma)$ over the time interval $\sigma \in [0, t_f]$ [178].

For a nonlinear system, observability hinges on the change of the output homeomorphism function as the state flows along the vector field of \dot{x} [167]. Succinct analysis of nonlinear observability utilizes the Lie derivative methodology; a thorough documentation of the technique can be found in [167]. Standard nonlinear form of system dynamics and the output equation in terms of the battery model are given as follows,

$$\dot{x}(t) = f(x, u) = \begin{pmatrix} -x_1/R_P C_D \\ -\Gamma(x_2)/q_0 R_D \end{pmatrix} + \begin{pmatrix} 1/C_D \\ -1/q_0 \end{pmatrix} u \quad (\text{A.14})$$

$$y(t) = h(x, u) = \Gamma(x_2) + x_1 + R_I u \quad (\text{A.15})$$

where $x_1 = V_D$, $x_2 = \bar{q}$, and $u = I_L$. The Lie derivative methodology takes subsequent vector derivatives of the above equations to determine the accessibility of the internal model states through the output manifold. To determine the observability, the Lie derivative is taken recursively the number of times equivalent to one plus the number of internal states [167]. For the battery system, given by equations (2.22) and (2.23), the Lie derivatives are calculated

as follows,

$$\begin{aligned}
L_f^0(h) &= h \\
L_f^1(h) &= \frac{\partial h}{\partial x} f \\
L_f^2(h) &= \frac{\partial}{\partial x} \left[\frac{\partial h}{\partial x} f \right] f = \frac{\partial}{\partial x} \left[L_f^1(h) \right] f
\end{aligned} \tag{A.16}$$

Execution of the methodology of equation (A.16) on the battery model in equations (A.14) and (A.15) yields the following observability space,

$$G_X = \begin{bmatrix} L_f^0(h) \\ L_f^1(h) \\ L_f^2(h) \end{bmatrix} = \begin{bmatrix} \frac{1}{C_D R_P} x_1 - \frac{\Gamma(0)}{q_0 R_D} \Gamma(1) \Gamma(0) \\ -\frac{1}{C_D^2 R_P^2} x_1 + \frac{1}{q_0^2 R_D^2} \left[\Gamma(1)^2 \Gamma(0) + \Gamma(2) \Gamma(0) \right] \end{bmatrix} \tag{A.17}$$

where the following nomenclature is adopted for derivatives of the discharge curve function, $\Gamma(x)$; $\Gamma(0) = \Gamma(x_2)$, $\Gamma(1) = \partial\Gamma(x_2)/\partial x_2$, and $\Gamma(2) = \partial^2\Gamma(x_2)/\partial x_2^2$. To determine the local observability of the battery model, the gradient of equation (A.17) must be computed. Observability is guaranteed given full rank of the following matrix,

$$\nabla G_X = \begin{bmatrix} -1 & \Gamma(1) \\ \frac{1}{C_D R_P} & -\frac{1}{q_0 R_D} \left[\Gamma(1)^2 \Gamma(0) + \Gamma(2) \Gamma(0) \right] \\ -\frac{1}{C_D^2 R_P^2} & \frac{1}{q_0^2 R_D^2} \left[\Gamma(1)^3 + 4\Gamma(2) \Gamma(1) \Gamma(0) + \Gamma(3) \Gamma(0) \right] \end{bmatrix} \tag{A.18}$$

Since $x_2 \in [0, 1]$ and $\Gamma(x_2) \in \mathbb{R}^+$ over this interval, the matrix in equation (A.18) exhibits full rank except when the battery is fully discharged ($x_2 = 0$). Further inspection of equation (A.18) illustrates the issue with observability of Li-ion battery chemistries. Li-ion cells, during discharge, exhibit an especially level complanate region. Resultantly, the $\Gamma(x_2)$ derivative functions in equation (A.18) will correspondingly shrink. Consequently, estimation become problematic in the complanate region as a result of error amplification.

Appendix B

DaNI Unmanned Ground Vehicle Experimental Results

Table B.1: Discharge process characteristics for the DaNI UGV experiments.

Exp. #	Runtime [min]	Power Load [W]	Max Current [A]	Shutdown Voltage [V]
1	64.59	15.95 ± 2.41	2.59	0.86
2	61.89	15.78 ± 2.27	2.75	0.87
3	68.21	15.56 ± 2.16	2.77	0.86
4	64.18	16.21 ± 2.42	2.34	0.87
5	61.72	14.99 ± 2.28	4.02	0.86
6	70.20	14.75 ± 1.98	2.60	0.88
7	70.47	14.15 ± 1.87	2.55	0.84
8	74.96	14.72 ± 1.93	2.37	0.89
9	69.73	14.16 ± 1.90	2.47	0.84
10	71.14	15.23 ± 1.82	3.20	0.87
11	66.13	14.39 ± 1.92	2.71	0.87
12	68.92	15.48 ± 2.14	2.74	0.93
13	66.70	14.69 ± 1.88	2.82	0.90
14	75.24	15.43 ± 1.97	3.34	0.89
15	62.02	14.55 ± 1.71	2.78	0.92
16	71.29	15.20 ± 1.89	2.66	0.89
17	69.54	14.56 ± 1.97	2.45	0.89
18	76.03	15.07 ± 1.71	3.05	0.91
19	72.98	15.10 ± 1.89	3.20	0.89
20	72.60	14.03 ± 1.91	2.65	0.86
21	73.06	14.41 ± 1.84	2.63	0.85
22	70.93	13.89 ± 1.82	2.58	0.90

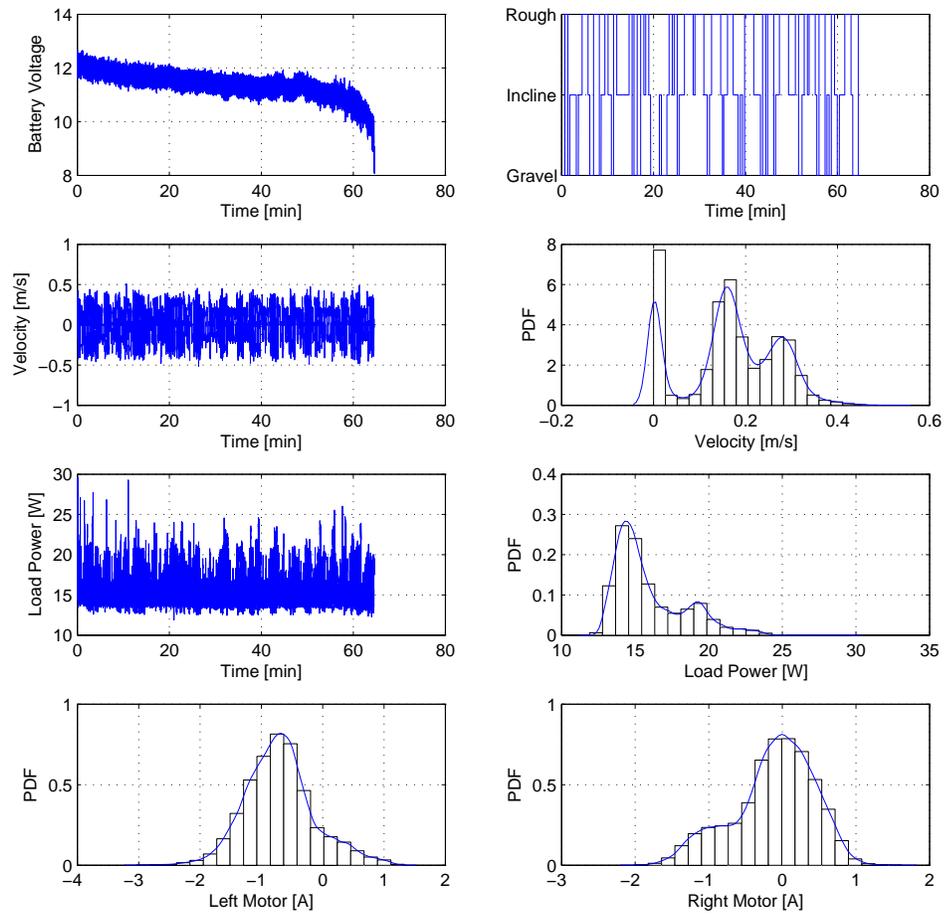


Figure B.1: DaNI UGV Experiment # 1.

Table B.2: Drive process characteristics for the DaNI UGV experiments.

Exp. #	Runtime [min]	Velocity [m/s]	Cumulative Distance [m]
1	64.59	0.165 ± 0.104	641.17
2	61.89	0.161 ± 0.100	598.22
3	68.21	0.159 ± 0.099	652.67
4	64.18	0.160 ± 0.110	616.68
5	61.72	0.159 ± 0.111	589.25
6	70.20	0.163 ± 0.106	686.13
7	70.47	0.158 ± 0.104	668.23
8	74.96	0.159 ± 0.106	713.48
9	69.73	0.155 ± 0.109	648.19
10	71.14	0.158 ± 0.107	672.22
11	66.13	0.158 ± 0.109	625.11
12	68.92	0.155 ± 0.107	640.40
13	66.70	0.151 ± 0.102	606.40
14	75.24	0.152 ± 0.106	685.92
15	62.02	0.151 ± 0.105	562.95
16	71.29	0.162 ± 0.103	690.78
17	69.54	0.162 ± 0.103	675.15
18	76.03	0.157 ± 0.101	715.27
19	72.98	0.155 ± 0.104	679.59
20	72.60	0.162 ± 0.109	706.72
21	73.06	0.158 ± 0.106	690.50
22	70.93	0.155 ± 0.105	659.65

Table B.3: In-house UGV study terrain distribution percentages.

Exp. #	Gravel [%]	Incline [%]	Rough [%]	Idle [%]
1	16.23	25.45	37.06	21.26
2	12.89	24.48	42.12	20.51
3	12.68	21.49	45.33	20.50
4	13.72	31.10	29.97	25.22
5	16.06	28.53	28.76	26.64
6	15.72	26.37	35.20	22.71
7	12.35	24.55	40.63	22.47
8	13.77	24.87	37.86	23.51
9	0.38	19.67	40.73	39.21
10	19.39	17.20	38.96	24.46
11	17.81	22.61	33.68	25.90
12	15.63	22.45	36.15	25.77
13	11.01	22.15	42.42	24.42
14	11.83	24.58	37.29	26.30
15	12.94	21.26	40.12	25.67
16	12.95	26.77	38.61	21.67
17	14.53	24.93	39.00	21.54
18	10.06	26.27	41.53	22.13
19	12.32	24.79	38.62	24.28
20	17.06	26.54	31.79	24.62
21	14.99	24.09	36.31	24.60
22	12.32	25.23	37.58	24.86

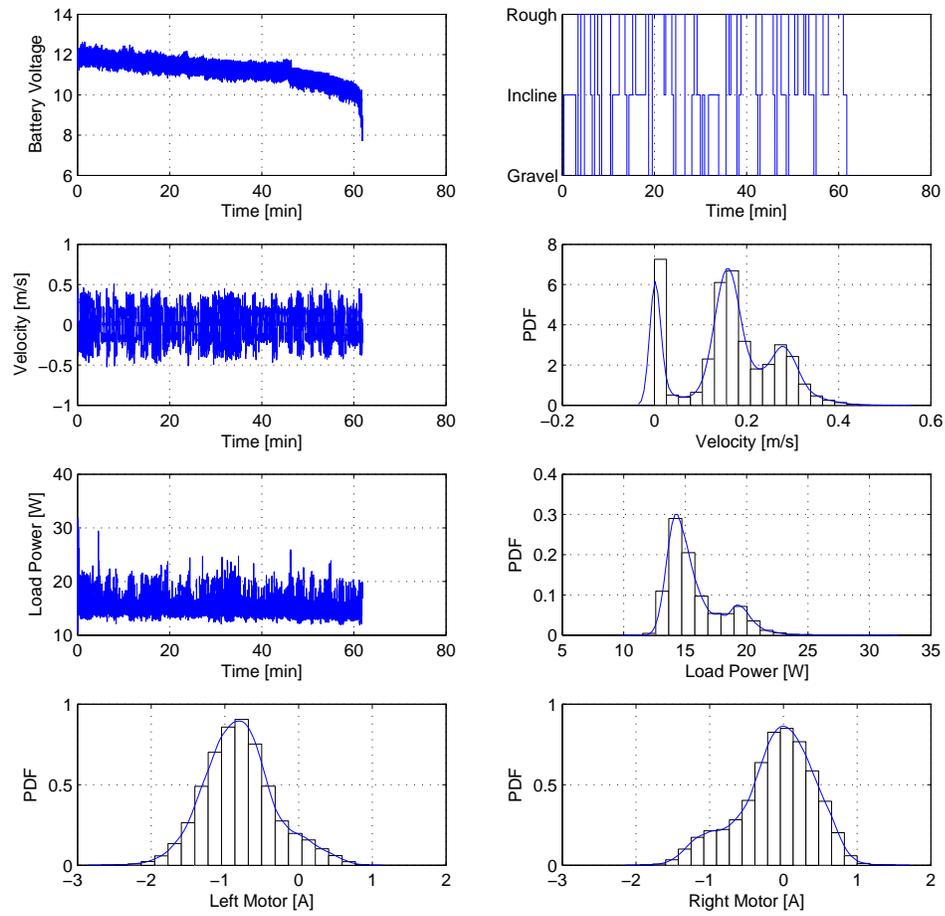


Figure B.2: DaNI UGV Experiment # 2.

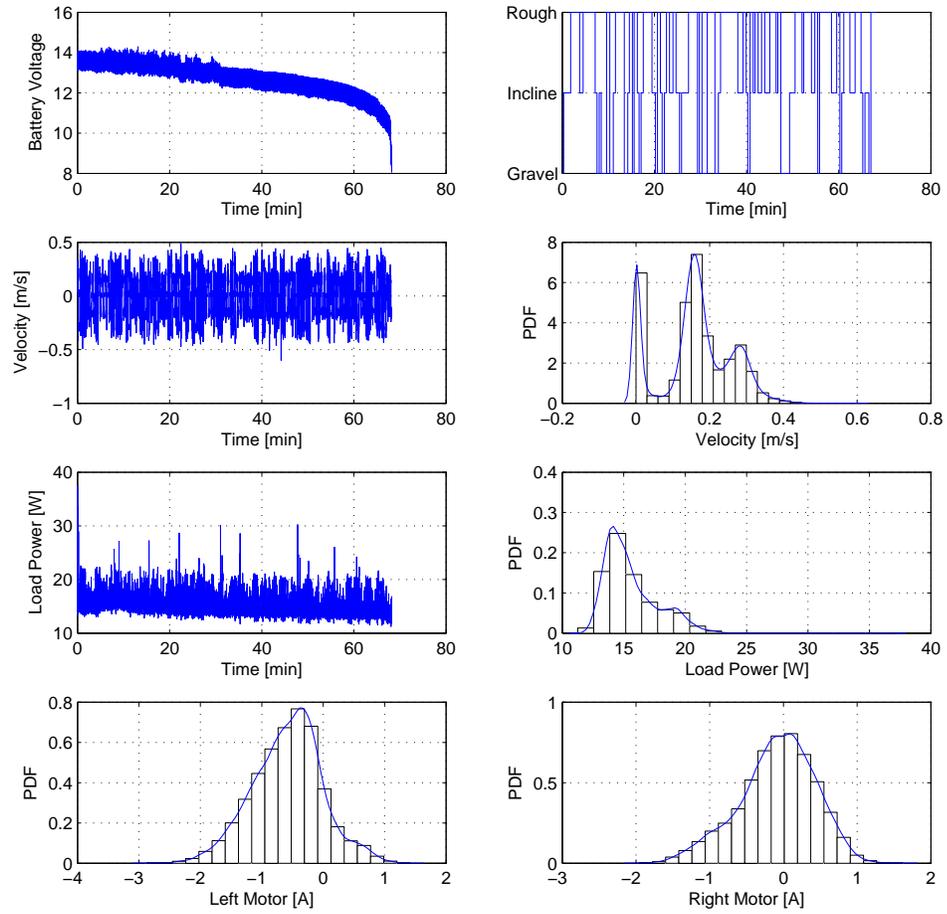


Figure B.3: DaNI UGV Experiment # 3.

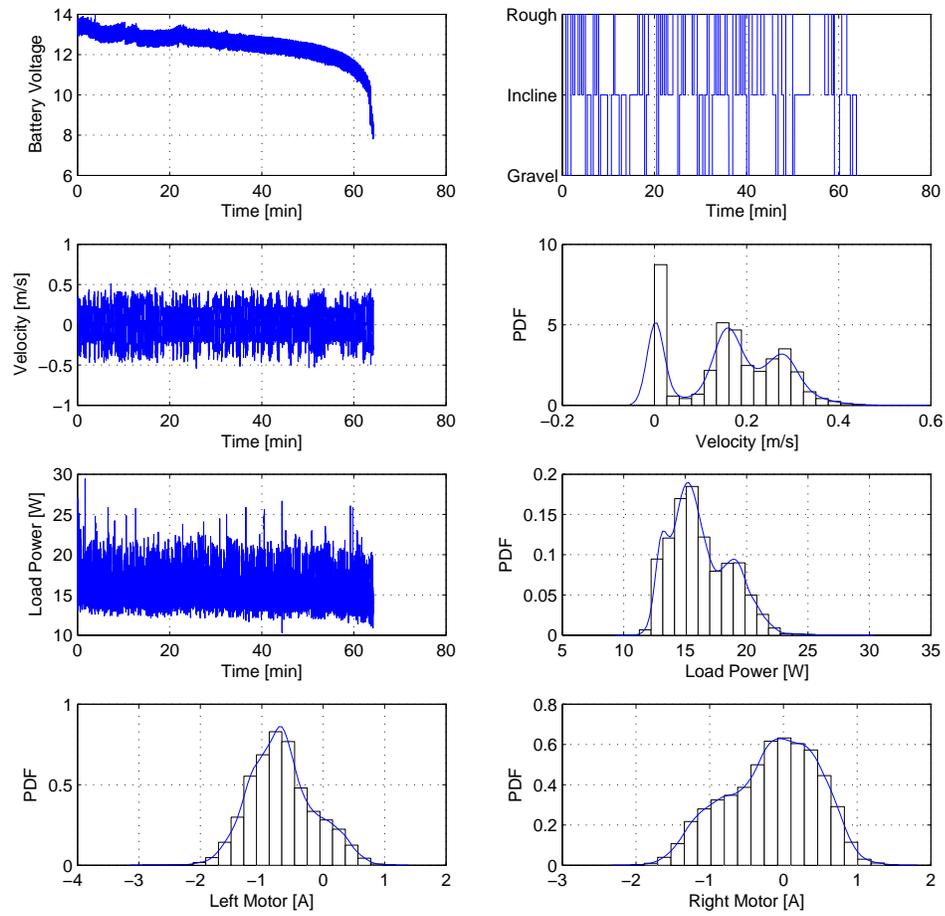


Figure B.4: DaNI UGV Experiment # 4.

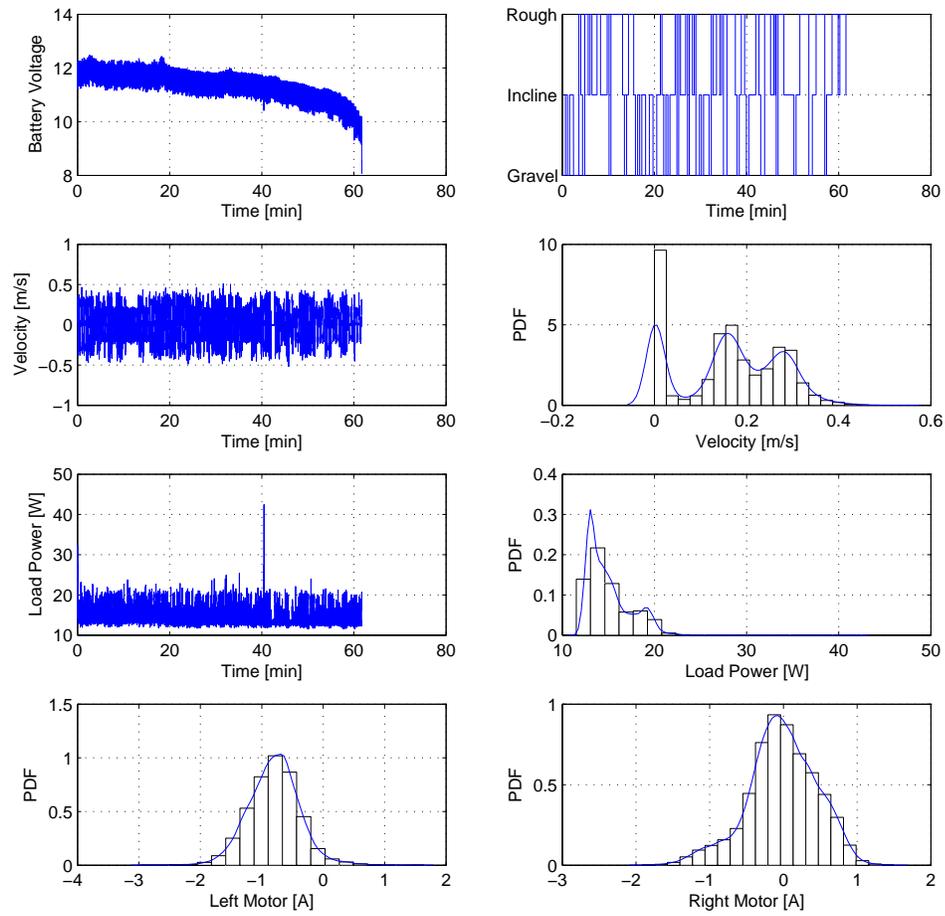


Figure B.5: DaNI UGV Experiment # 5.

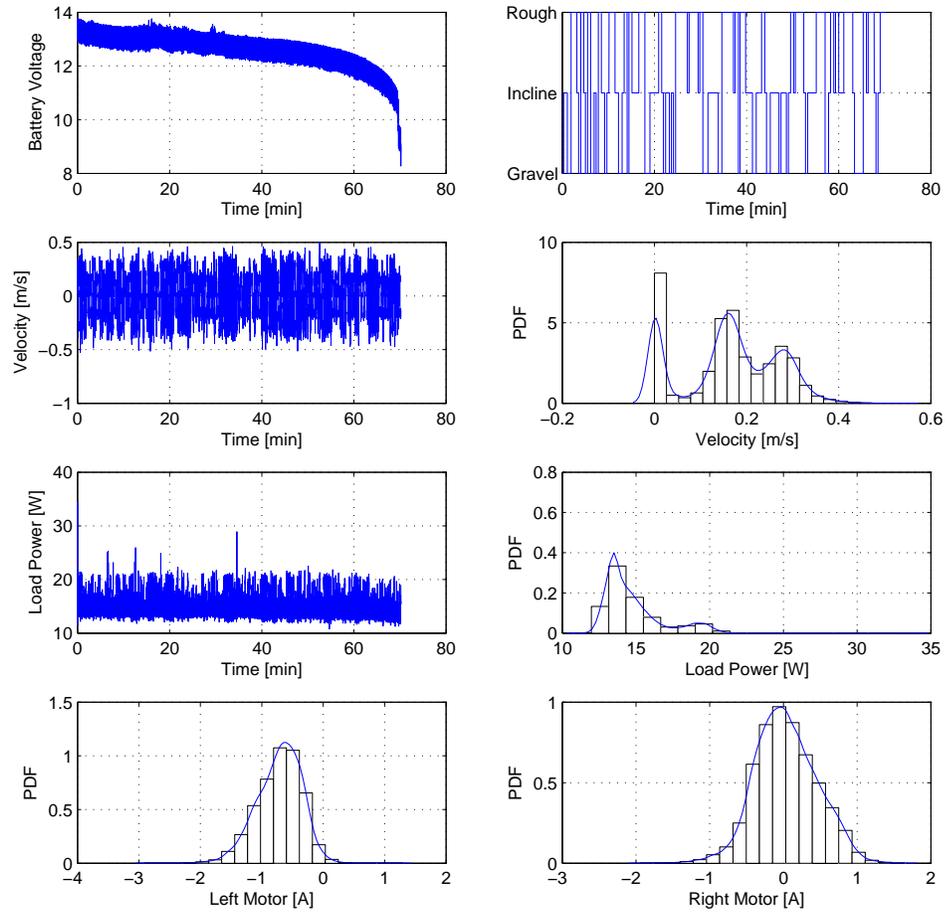


Figure B.6: DaNI UGV Experiment # 6.

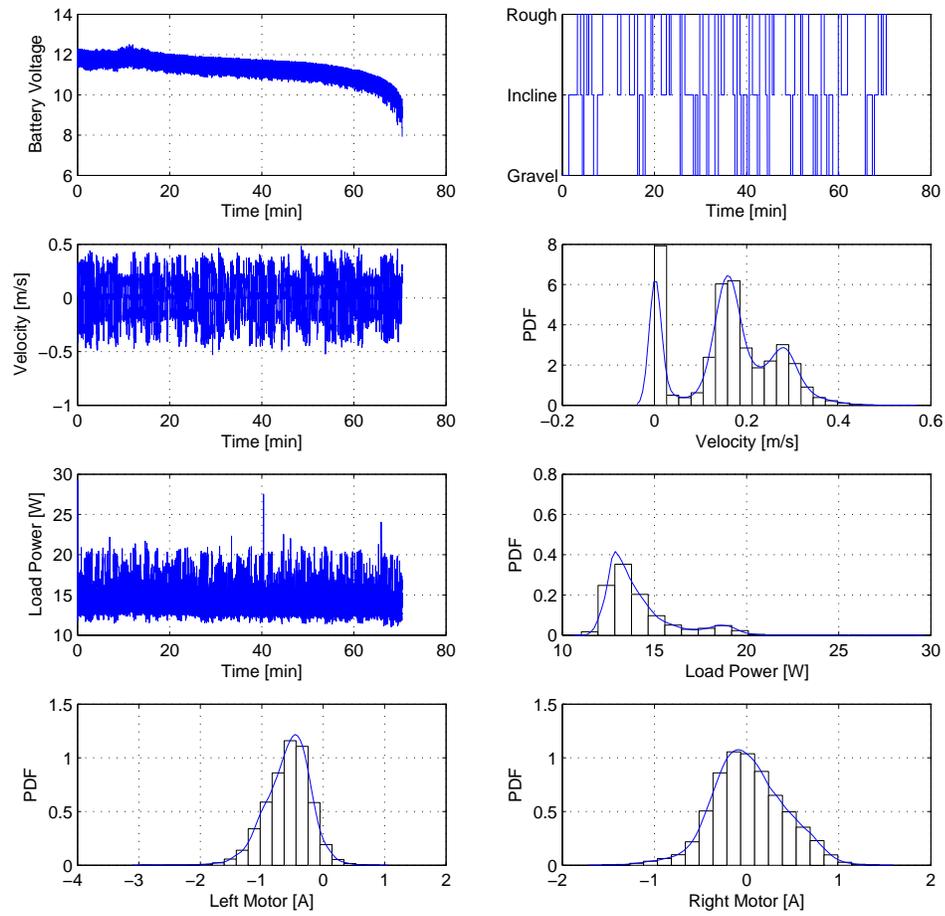


Figure B.7: DaNI UGV Experiment # 7.

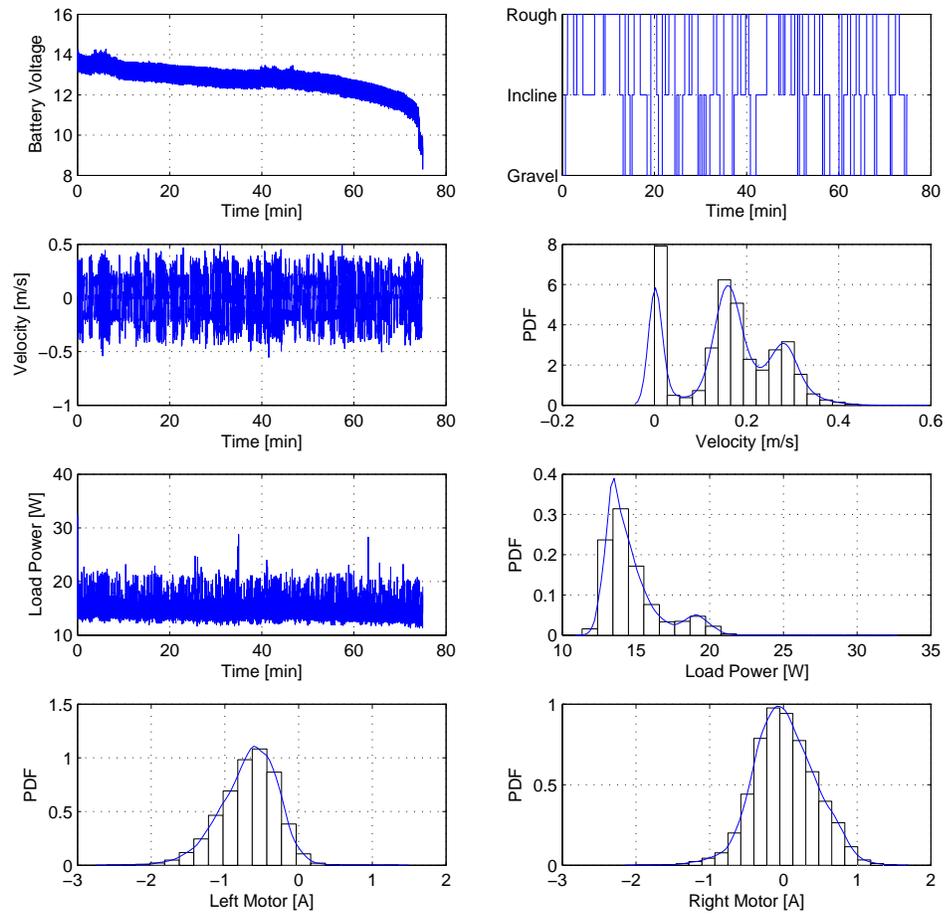


Figure B.8: DaNI UGV Experiment # 8.

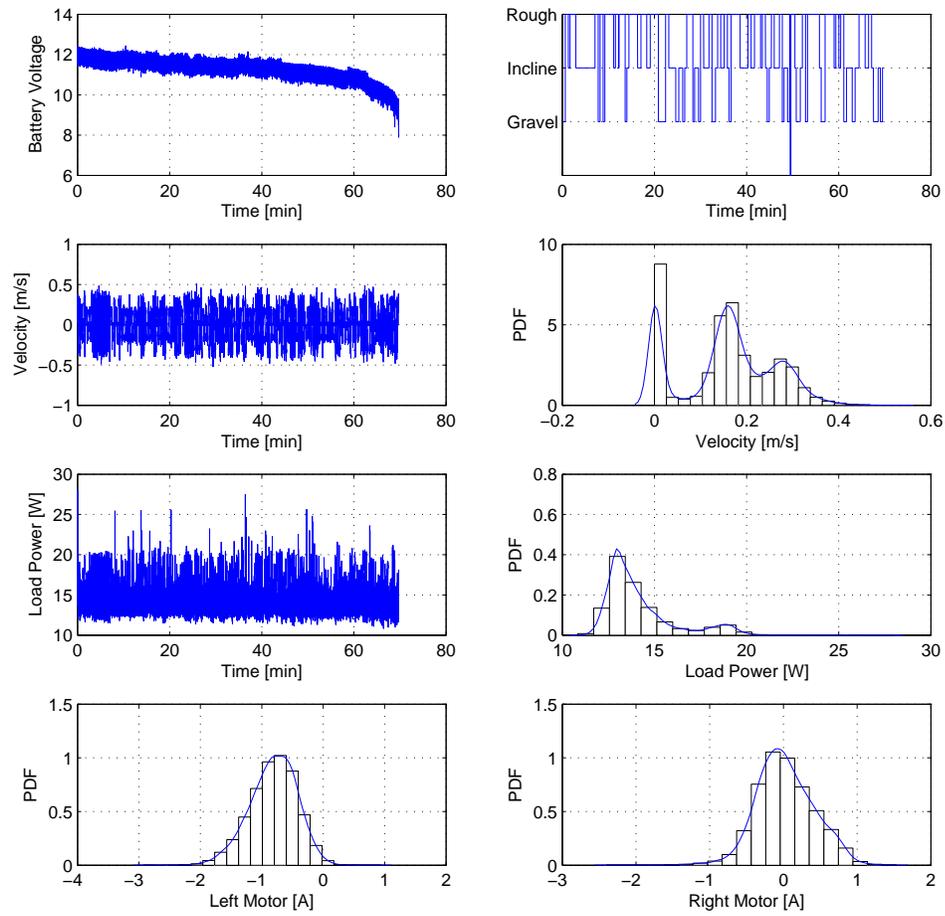


Figure B.9: DaNI UGV Experiment # 9.

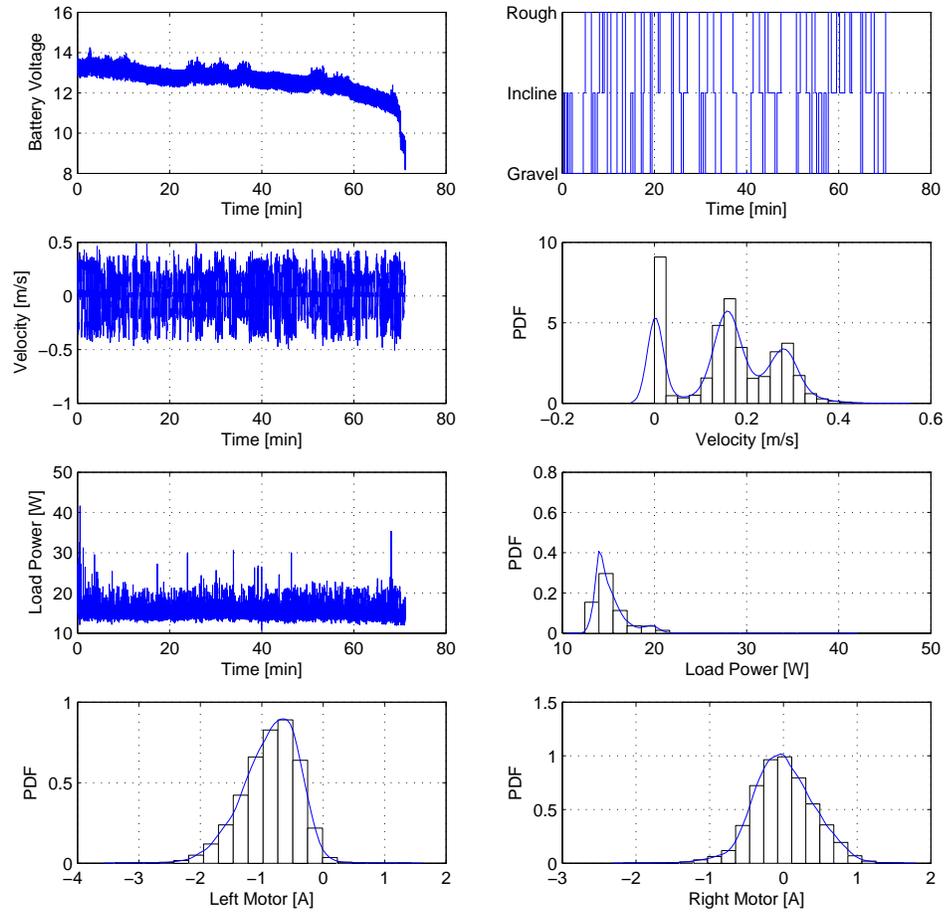


Figure B.10: DaNI UGV Experiment # 10.

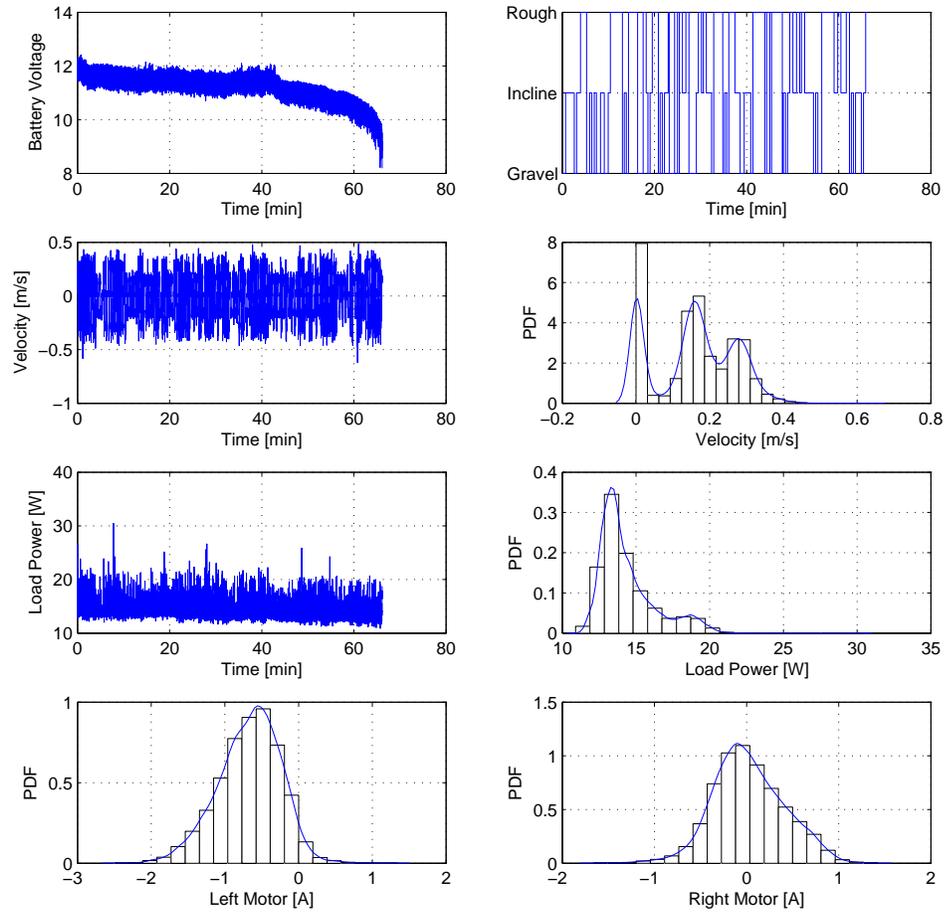


Figure B.11: DaNI UGV Experiment # 11.

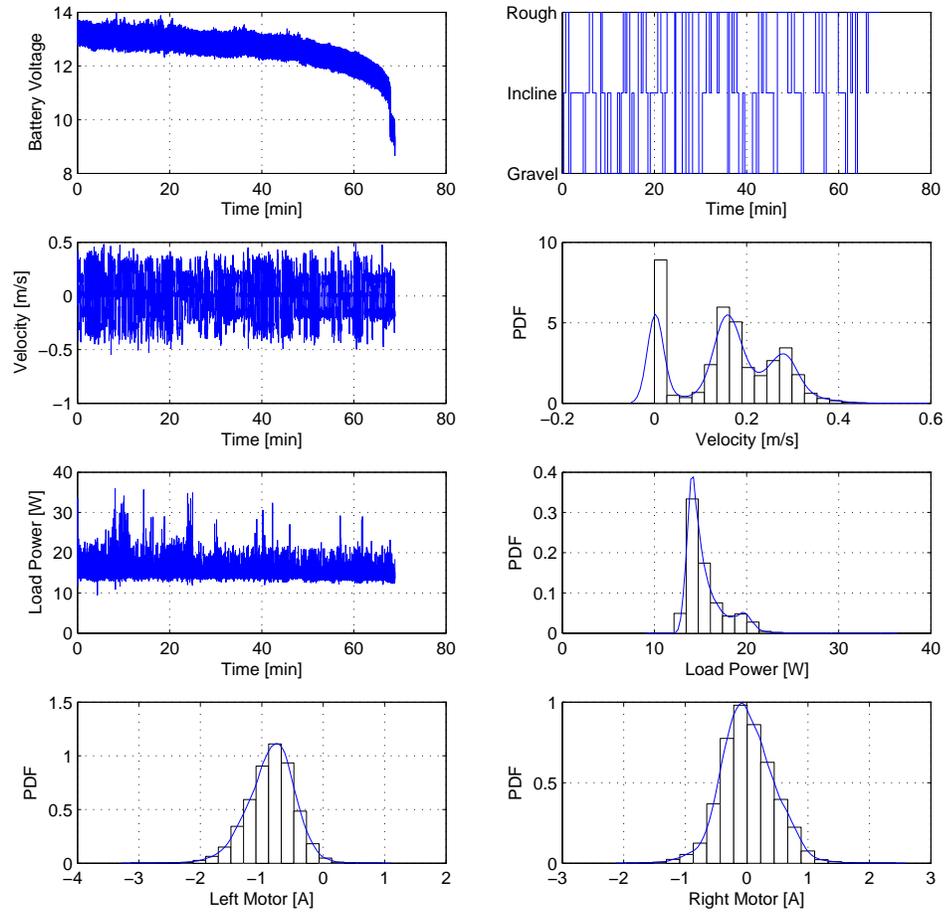


Figure B.12: DaNI UGV Experiment # 12.

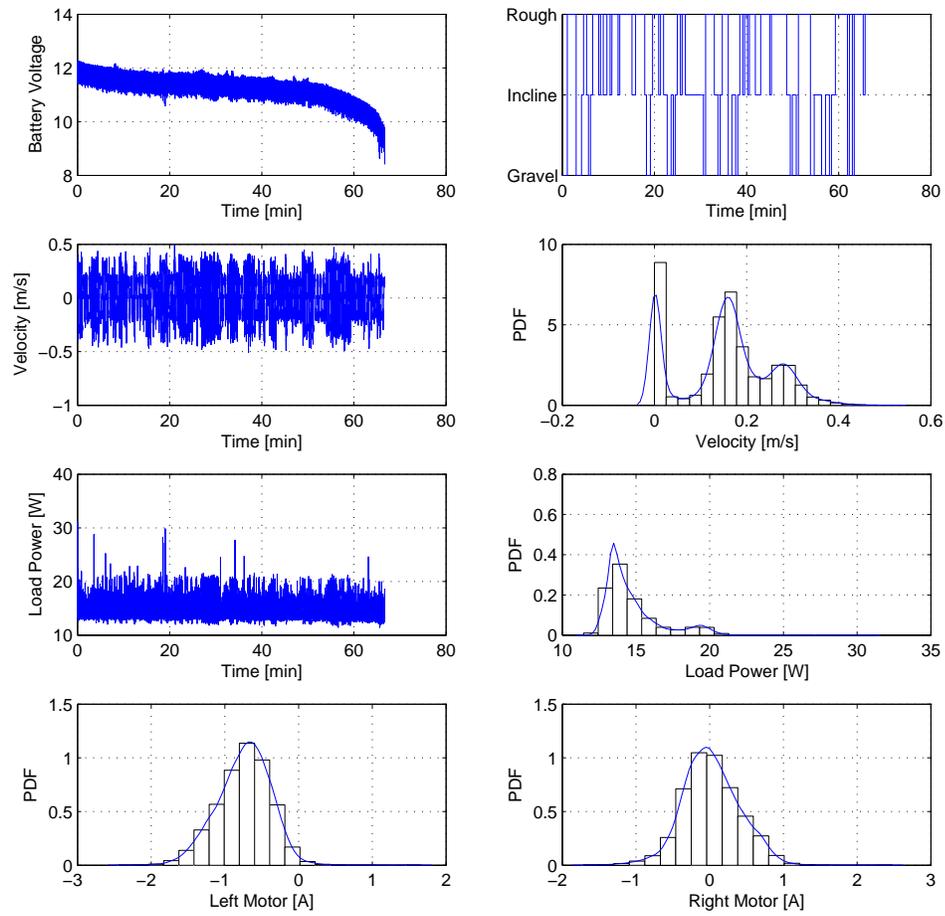


Figure B.13: DaNI UGV Experiment # 13.

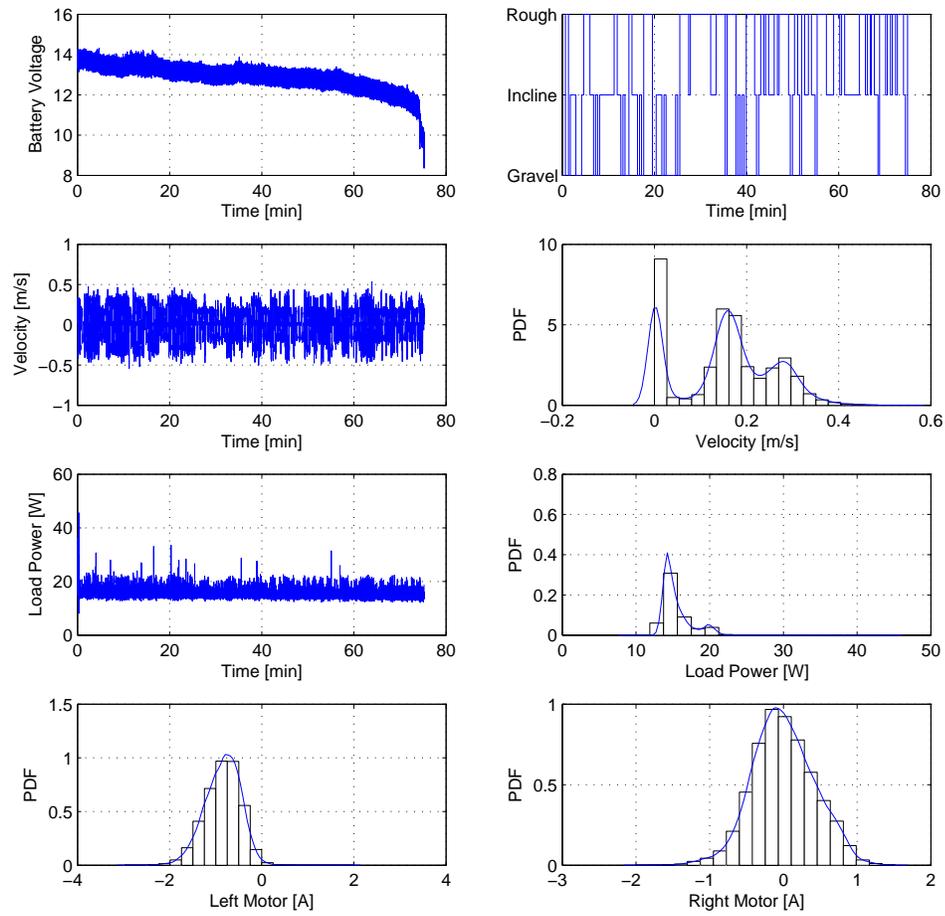


Figure B.14: DaNI UGV Experiment # 14.

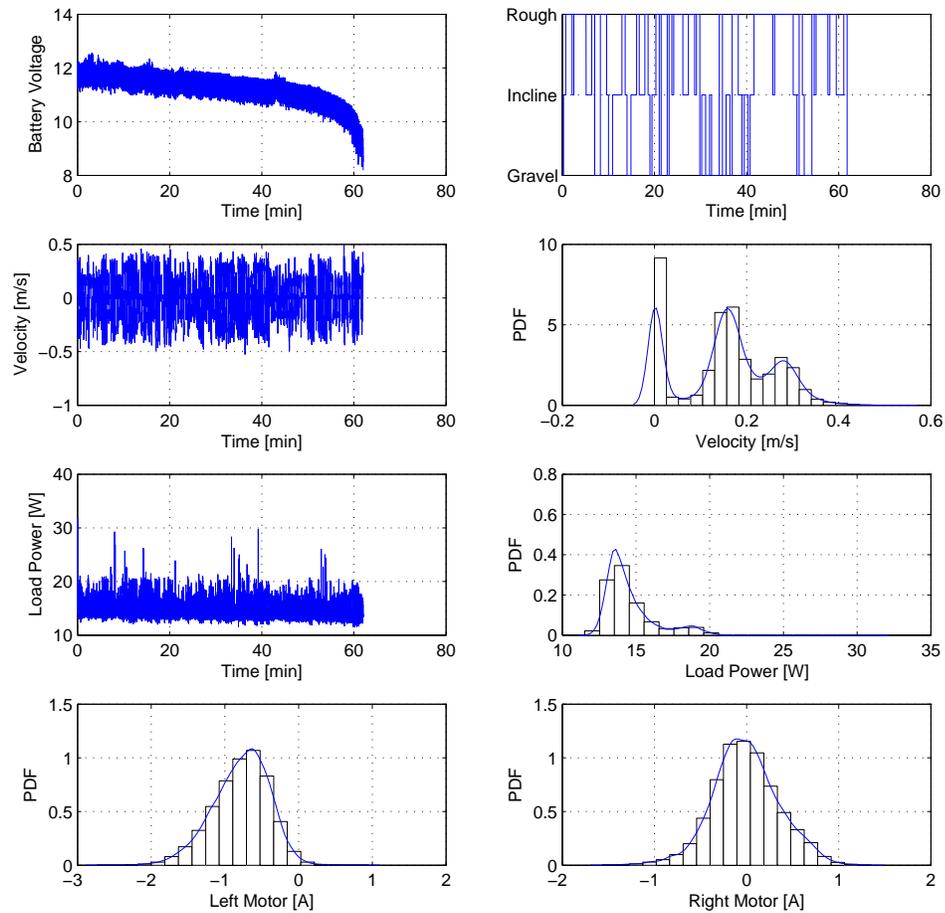


Figure B.15: DaNI UGV Experiment # 15.

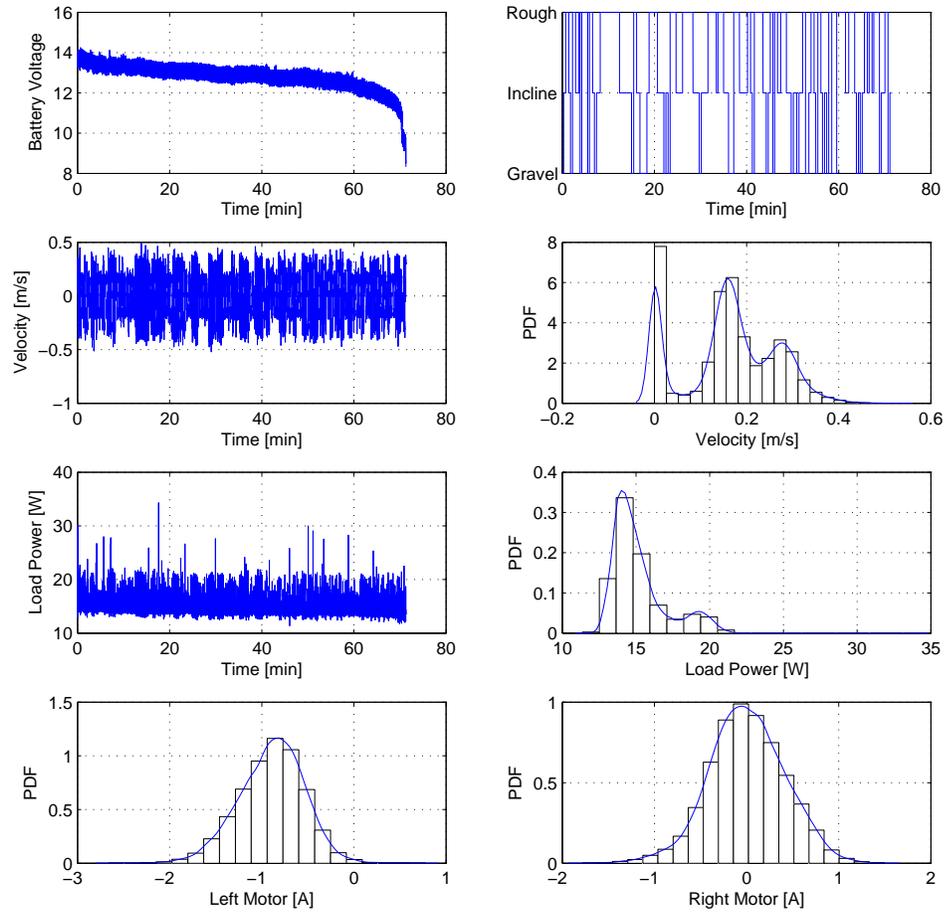


Figure B.16: DaNI UGV Experiment # 16.

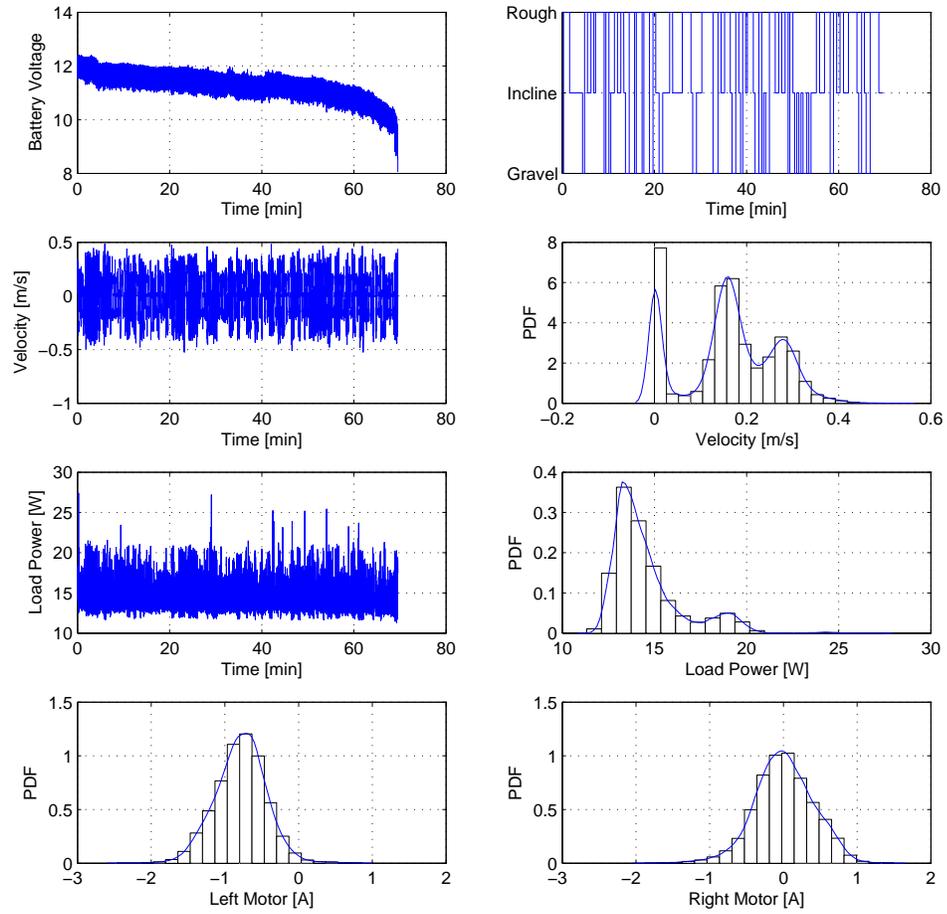


Figure B.17: DaNI UGV Experiment # 17.

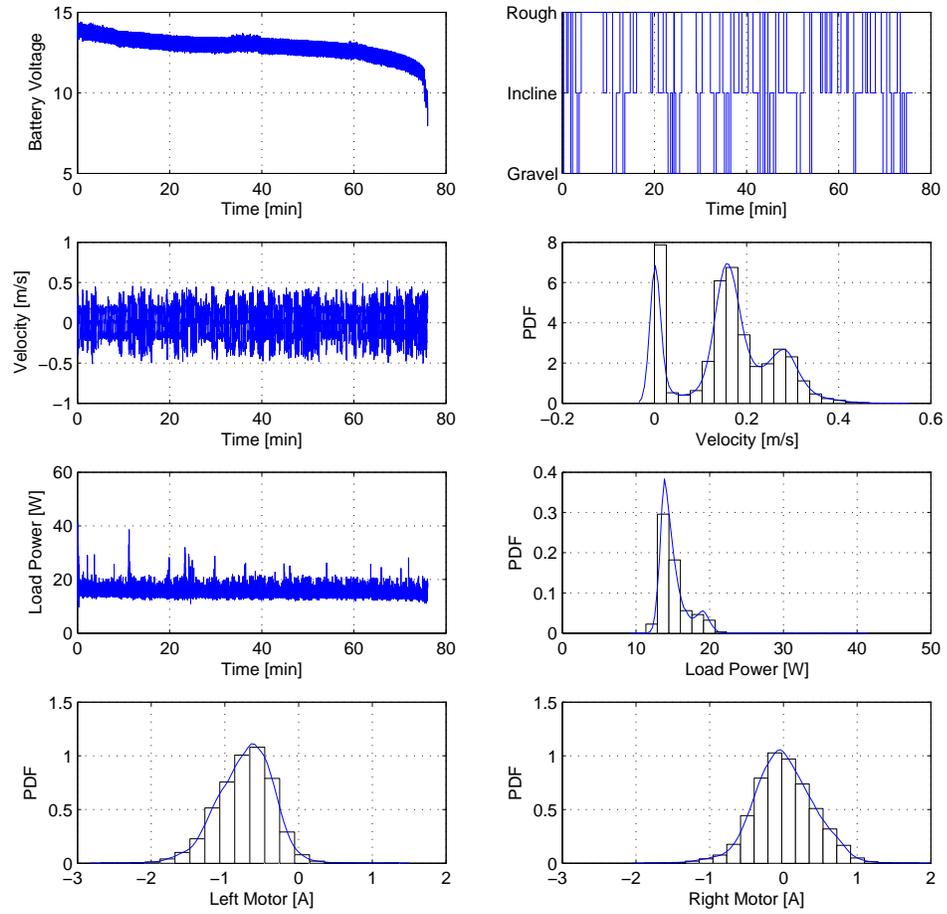


Figure B.18: DaNI UGV Experiment # 18.

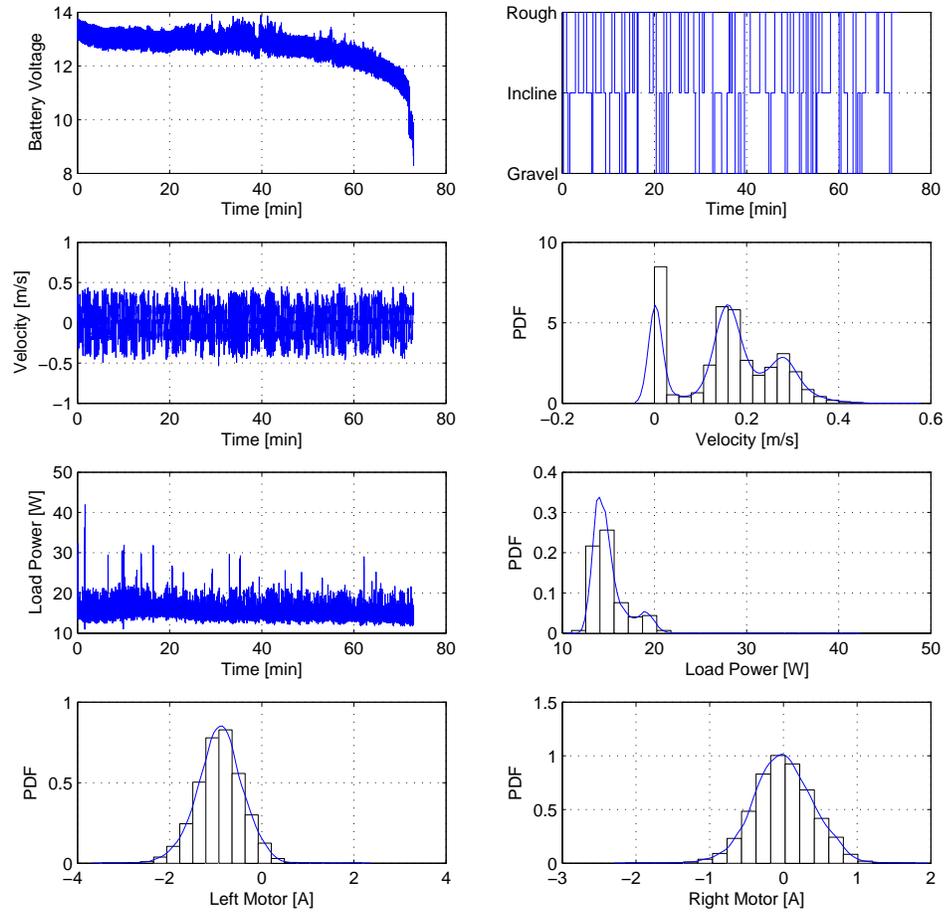


Figure B.19: DaNI UGV Experiment # 19.

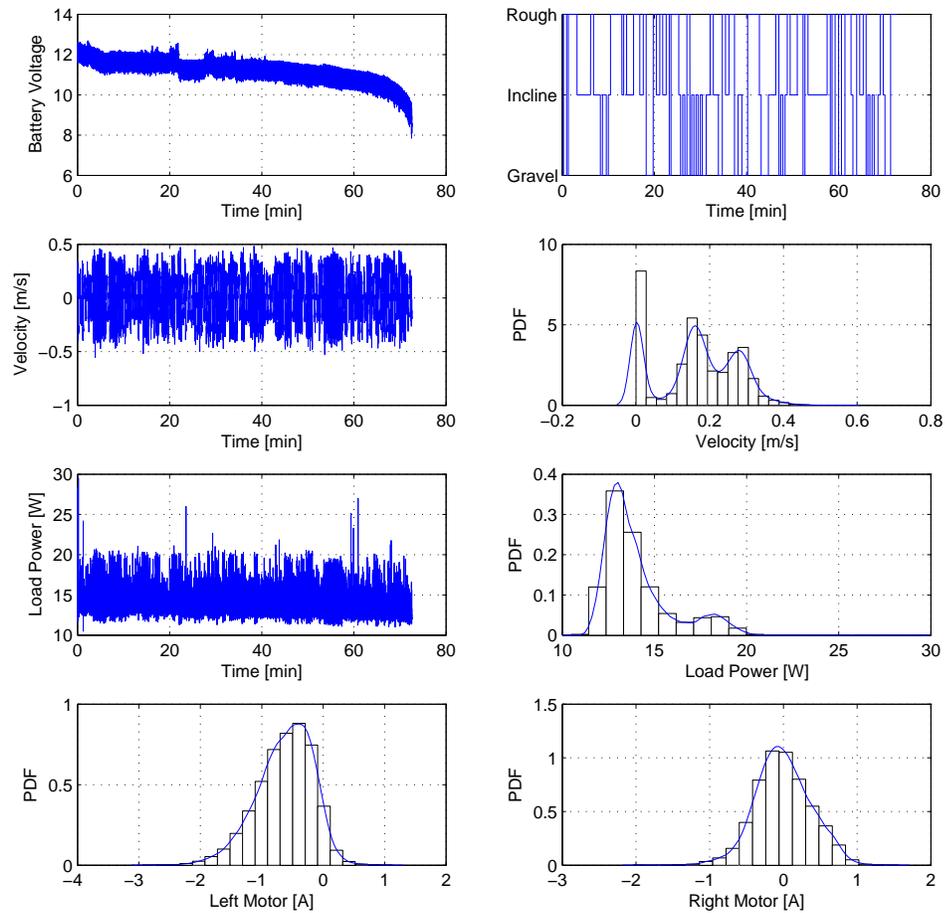


Figure B.20: DaNI UGV Experiment # 20.

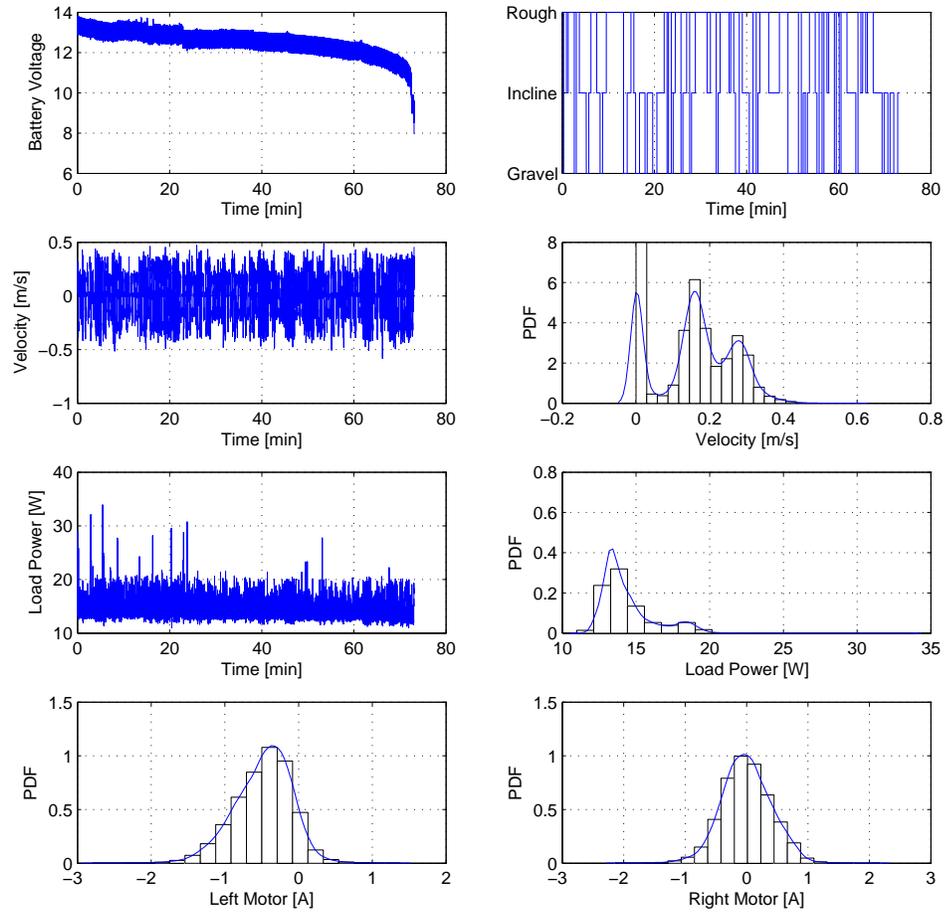


Figure B.21: DaNI UGV Experiment # 21.

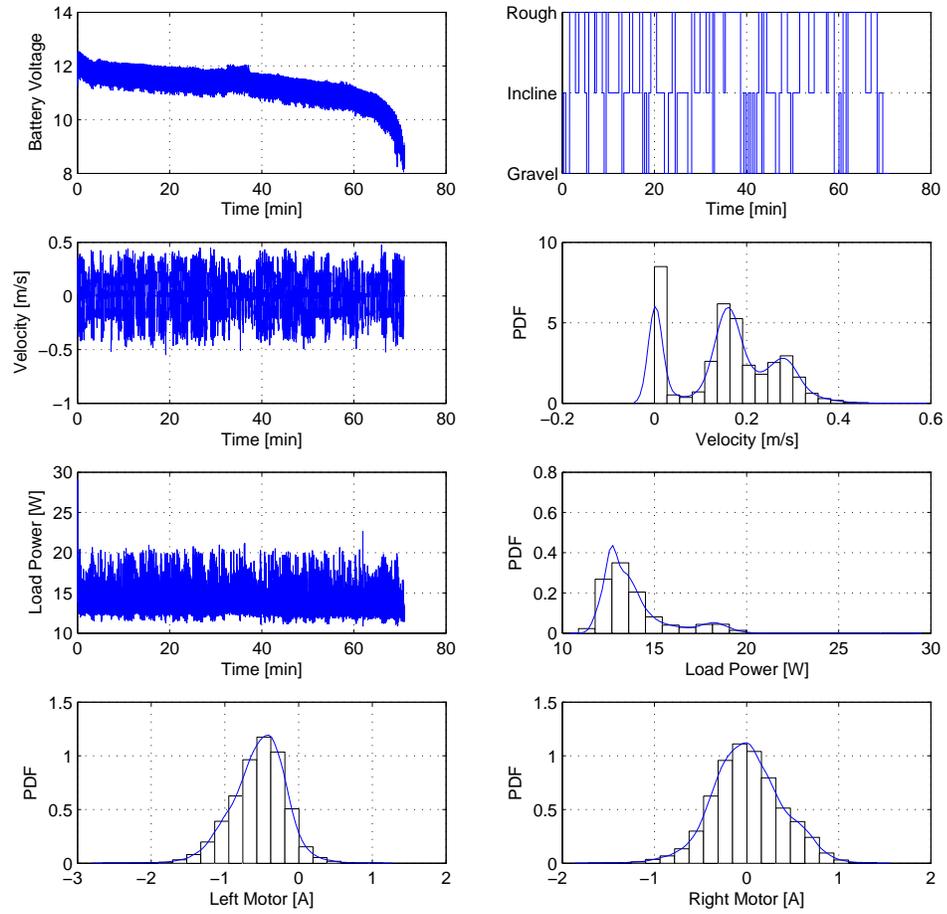


Figure B.22: DaNI UGV Experiment # 22.

Appendix C

Univariate Gaussian Mixtures and Clustering

Simply stated, Gaussian mixture modeling approximates non-Gaussian (potentially multi-modal) density functions via a convex combination of Gaussian densities. While density approximation alone is a potent numerical technique, data clustering via GMM provided new avenues of research and extended the practicality of the algorithm, notably in the field of machine learning. In this body of work, the GMM methodology is used to cluster load data (training data) into a priori unknown regions. In the following appendix, the definition of the GMM and associated properties are stated. Subsequently the expectation-maximization algorithm, which provides the parameters for the model, is discussed. Finally, the EM algorithm for GMMs specifically is detailed.

C.1 Statistical Mixture Model

Since clustering in this work is restricted to univariate models, the GMM density functions are presented in this section as univariate without loss of generality. The overall model density function can be stated simply as

follows,

$$P_{\theta}(x) = \sum_{k=1}^M \alpha_k \phi(x|\mu_k, \sigma_k^2) \quad (\text{C.1})$$

where the model parameters to be found are $\hat{\theta} = [\hat{\alpha}_k, \hat{\mu}_k, \hat{\sigma}_k^2]$, the distributions of $\phi(x|\mu_k, \sigma_k^2)$ are normally distributed, and α_k are mixture weights of the Gaussian distributions which must satisfy the normalizing condition, $\sum_{k=1}^M \alpha_k = 1$. However, to exploit mathematical properties of logarithms eventually, the GMM is usually stated in joint multiplicative form as opposed to the additive form of equation (C.1) as follows,

$$P_{\theta}(x) = \prod_{k=1}^M \alpha_k^{z_k} \phi(x|\mu_k, \sigma_k^2)^{z_k} \quad (\text{C.2})$$

where the parameters θ and data vector X remain the same. The distribution is now expressed jointly with Z , which is a unit vector which chooses which mixture model is currently expressed. As such, elements of Z belong to the set 0,1. For a given a data vector $X = x_1, x_2, \dots, x_n$, the likelihood function defines fit fidelity, specifically the likelihood of the data X occurring given the parameters $\hat{\theta}$. To exploit logarithmic identities, commonly the log-likelihood function is optimized.

$$L(\theta|X) = \ln \prod_{k=1}^M \alpha_k^{z_k} \phi(x|\mu_k, \sigma_k^2)^{z_k} = \sum_{k=1}^M \ln \alpha_k^{z_k} \phi(x|\mu_k, \sigma_k^2)^{z_k} \quad (\text{C.3})$$

Additionally, since the logarithm is a monotone transform, the log-likelihood function preserves the optima of the likelihood. Succinctly stated, to find parameters X , one must optimize the log-likelihood function as follows,

$$\theta_{MLE} \in \arg \max_{\theta \in \Theta} L(\theta|X) \quad (\text{C.4})$$

Solving the optimization problem in equation (C.4) for GMMs is analytically intractable. Resultantly, one must employ a numerical methodology to compute the parameters. While numerical techniques, such as Markov chain Monte Carlo and moment matching have been used to fit GMMs, the expectation maximization (EM) algorithm is overwhelmingly preferred. In practical problems (low-moderate dimensionality and cluster number), the EM algorithm demonstrates robustness and quick convergence.

C.2 Expectation-Maximization Algorithm

The standard technique for computing the maximum likelihood parameters is the expectation maximization algorithm. The EM algorithm iteratively calculates successively more likely parameters for the GMM until convergence. Consequently, the algorithm does not guarantee convergence to the global optimum due to the lack of convexity. However, mixture model fidelity is commonly ensured via judiciously selected initial parameter estimates and via cluster validation metrics, such as cluster cohesion and cluster separation. Iterations through the EM algorithm are comprised of two steps. Firstly in the so called expectation step or "E-step", the expected value of the log-likelihood equations is compute, shown in equation (C.5). One caveat to note, the general formulation of the EM algorithm includes a general hidden parameter, Z . In the case of the GMM formulation, these Z parameters will indicate the probability of each data point belonging to each cluster (so called

soft-clustering).

$$Q(\theta, \theta_t) = E(\ln P_\theta(X, Z) | \theta_t, X = x) \quad (\text{C.5})$$

where θ_t is the previous iterative value of the parameters. Secondly, the computed expectation of the previous step is maximized, in the so called maximization step or “M-step”.

$$\theta_{t+1} \in \arg \max_{\theta \in \Theta} Q(\theta, \theta_t) \quad (\text{C.6})$$

The above equation computes the θ which is used during the next iterative step. [138] illustrates the proof demonstrating the maximization properties of the above algorithm. To compute the parameters for the GMM via the EM algorithm, the log-likelihood function in equation (C.3) must be used in the E-step. After ample manipulation, the expectation step equation reduces to a relationship, see equation (C.7), which resembles Bayes rule. During each step of the EM algorithm, the E-step computes the respective probability of each data point belong to each mixture model. Since each data point exhibits probability of belonging to each cluster, the E-step is termed a soft clustering step.

$$P(C_m | x_i) = \frac{\hat{\alpha}_m \phi(x_i | \hat{\theta}_t)}{\sum_{m=1}^M \hat{\alpha}_m \phi(x_i | \hat{\theta}_t)} \quad (\text{C.7})$$

Once the relative probabilities of the data points have been computed, the log likelihood function must be optimized. The most straightforward method for determining the EM algorithm recursive relationships can be derived by maximizing the Q function generally and ensuring that the significant statistics

of the GMM model obey the recursion [85]. The maximum of the expectation function can be found as follows,

$$\frac{\partial}{\partial \theta_i} Q = E_{\theta_0}(S_i(X, Z|X = x) - E_{\theta}(S_i(X, Z)) = 0 \quad (\text{C.8})$$

Enforcing the above condition for the significant statistics of equation (C.2), ensures that each successive iteration of the EM algorithm with increase the maximum likelihood function. For the univariate case, the update equation for the mixture weights is found to be,

$$\alpha_k = \frac{1}{N} \sum_{j=1}^N P(C_k|x_j, \theta_t) \quad (\text{C.9})$$

where N is the total number of data point and C_k are the current clusters. In a similar fashion, the update equations for μ and σ can be computed as follows,

$$\mu_k = \frac{\sum_{j=1}^N x_j P(C_k|x_j, \theta_t)}{\sum_{j=1}^N P(C_k|x_j, \theta_t)} \quad (\text{C.10})$$

$$\sigma_k = \frac{\sum_{j=1}^N [(x_j - \mu_k)(x_j - \mu_k)^T] P(C_k|x_j, \theta_t)}{\sum_{j=1}^N P(C_k|x_j, \theta_t)} \quad (\text{C.11})$$

The code written by the author which implements the above algorithm can be found for download at ”<https://github.com/chromodynamic/mark-meets-gauss/blob/master/fitGMM.m>” or by contacting the author through the listed contact information.

Bibliography

- [1] I. Hadjipaschalis, A. Poullikkas, and V. Efthimiou, “Overview of current and future energy storage technologies for electric power applications,” *Renewable and Sustainable Energy Reviews*, vol. 13, no. 6-7, pp. 1513–1522, 2009.
- [2] S. Jaffe, C. Talon, K. Ishimori, and R. Nicholson, “Business Strategy: Lithium Ion Manufacturing Global Buildout Supply and Demand Forecasts,” tech. rep., IDC Energy Insights, Framingham, MA, 2001.
- [3] K. Boice, J. Paulson, T. Valascho, A. Leo, and J. Lee, “Baseline Field Testing of BB-2590 Lithium-Ion Batteries using an iRobot FasTac 510 Robot,” Tech. Rep. 21320, US Army, Tank Automotive Research Development and Engineering Command, Warren, MI, 2010.
- [4] L. E. Ray, J. H. Lever, N. Hampshire, A. D. Streeter, and A. D. Price, “Design and Power Management of a Solar-Powered Cool Robot for Polar Instrument Networks,” *Journal of Field Robotics*, vol. 24, no. 7, pp. 581–599, 2007.
- [5] R. Hebner, J. Herbst, and A. Gattozzi, “Large scale simulations of a ship power system with energy storage and multiple directed energy loads,” in ... *Challenges in Modeling & Simulation* ..., pp. 2–7, 2010.

- [6] J. V. C. Vargas, J. A. Souza, R. Hovsapian, J. C. Ordonez, J. Chalfant, C. C. Esrdc, and C. Gcms, “ESRDC ship notional baseline Medium Voltage Direct Current (MVDC) architecture thermal simulation and visualization,” in *Grand Challenges on Modeling and Simulation Conference*, no. Mvdc, (Vista, CA), pp. 150–160, 2011.
- [7] D. Linden, *Handbook of batteries*. New York: McGraw Hill, 2 ed., 2005.
- [8] D. G. Logan, J. Pentzer, S. N. Brennan, and K. Reichard, “Comparing batteries to generators as power sources for use with mobile robotics,” *Journal of Power Sources*, vol. 212, pp. 130–138, Aug. 2012.
- [9] C. R. Gould, C. M. Bingham, D. A. Stone, and P. Bentley, “New Battery Model and State-of-Health Determination Through Subspace Parameter Estimation and State-Observer Techniques,” *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 8, pp. 3905–3916, 2009.
- [10] D. Bernardini and A. Bemporad, “Energy-aware robust model predictive control based on noisy wireless sensors,” *Automatica*, vol. 48, pp. 36–44, Jan. 2012.
- [11] B. S. K. Goebel, “Model Adaptation for Prognostics in a Particle Filtering Framework,” *International Journal of Prognostics and Health Management*, 2011.
- [12] C. Lundberg, H. I. Christensen, and R. Reinhold, “Long-Term Study of a Portable Field Robot in Urban Terrain,” *Journal of Field Robotics*,

vol. 24, pp. 625–650, 2007.

- [13] B. Saha, E. Koshimoto, C. C. Quach, E. F. Hogge, T. H. Strom, B. L. Hill, S. L. Vazquez, and K. Goebel, “Battery health management system for electric UAVs,” in *Aerospace Conference, 2011 IEEE*, pp. 1–9, Mar. 2011.
- [14] W. Peukert, “Elektrotechnische Zeitschrift,” no. 20, pp. 20–21, 1897.
- [15] M. Dalal, J. Ma, and D. He, “Lithium-ion battery life prognostic health management system using particle filtering framework,” *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, vol. 225, pp. 81–90, Mar. 2011.
- [16] C.-C. Lin, H. Peng, and J. W. Grizzle, “A stochastic control strategy for hybrid electric vehicles,” in *American Control Conference, 2004. Proceedings of the 2004*, vol. 5, pp. 4710–4715 vol.5, 2004.
- [17] J. LeSage and R. Longoria, “Characterization of Load Uncertainty in Unstructured Terrains and Applications to Battery Remaining Runtime Prediction,” *Journal of Field Robotics*, pp. 1–16, 2013.
- [18] C. D. Rahn, *Battery Systems Engineering*. New York: Wiley, 1 ed., 2012.
- [19] L. Cloth, M. R. Jongerden, and B. R. Haverkort, “Computing Battery Lifetime Distributions,” in *Proceedings of the 37th Annual IEEE/IFIP*

- International Conference on Dependable Systems and Networks*, DSN '07, (Washington, DC, USA), pp. 780–789, IEEE Computer Society, 2007.
- [20] M. R. Jongerden, *Model-based energy analysis of battery powered systems*. PhD thesis, University of Twente, Dec. 2010.
- [21] A. Sadrpour and A. G. Ulsoy, “Mission Energy Prediction for Unmanned Ground Vehicles Using Real-time Measurements and Prior Knowledge,” *Journal of Field Robotics*, vol. 30, no. 3, pp. 1–16, 2013.
- [22] C. Zhang and A. Vahidi, “Route Preview in Energy Management of Plug-in Hybrid Vehicles,” *IEEE Transactions on Control Systems Technology*, vol. 20, pp. 546–553, Mar. 2012.
- [23] V. Rykov, N. Balakrishnan, and M. Nikulin, *Mathematical and Statistical Models and Methods in Reliability: Applications to Medicine, Finance, and Quality Control*. Birkhauser, 2010.
- [24] G. V. F. L. L. M. R. A. H. B. Wu, *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. John Wiley and Sons, Inc., 2006.
- [25] R. González, M. Fiacchini, J. L. Guzmán, T. Álamo, and F. Rodríguez, “Robust tube-based predictive control for mobile robots in off-road conditions,” *Robotics and Autonomous Systems*, vol. 59, pp. 711–726, Oct. 2011.

- [26] J. Carlson and R. Murphy, “How UGVs physically fail in the field,” *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 423–437, 2005.
- [27] Y. Ding, “U.S. Armys Ground Vehicle Energy Storage R&D Programs & Goals,” tech. rep., Energy Storage Team, US Army TARDEC, Warren, MI, 2011.
- [28] D. Doerffel and S. A. Sharkh, “A critical review of using the Peukert equation for determining the remaining capacity of lead-acid and lithium-ion batteries,” *Journal of Power Sources*, vol. 155, no. 2, pp. 395–400, 2006.
- [29] D. Rakhmatov and S. Vrudhula, “Energy management for battery-powered embedded systems,” *ACM Trans. Embed. Comput. Syst.*, vol. 2, pp. 277–324, Aug. 2003.
- [30] J.-M. Kang, C.-K. Park, S.-S. Seo, M.-J. Choi, and J. W.-K. Hong, “User-Centric Prediction for Battery Lifetime of Mobile Devices,” in *Proceedings of the 11th Asia-Pacific Symposium on Network Operations and Management: Challenges for Next Generation Network Operations and Service Management*, APNOMS '08, (Berlin, Heidelberg), pp. 531–534, Springer-Verlag, 2008.
- [31] F. Kerasiotis, A. Prayati, C. Antonopoulos, C. Koulamas, and G. Papadopoulos, “Battery Lifetime Prediction Model for a WSN Platform,” in *Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on*, pp. 525–530, July 2010.

- [32] K. N. Truong, J. A. Kientz, T. Sohn, A. Rosenzweig, A. Fonville, and T. Smith, “The design and evaluation of a task-centered battery interface,” in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, Ubicomp ’10, (New York, NY, USA), pp. 341–350, ACM, 2010.
- [33] A. M. Pesco, R. V. Biagetti, R. S. Chidamber, and C. R. Venkatram, “An adaptive battery reserve time prediction algorithm,” in *Telecommunications Energy Conference, 1989. INTELEC ’89. Conference Proceedings., Eleventh International*, pp. 6.1/1 –6.1/7 vol.1, Oct. 1989.
- [34] T. Matsushima, S. Ishizuka, and M. Hashiwaki, “Prediction of remaining battery discharge time in telecommunications systems,” in *Telecommunications Energy Conference, 1990. INTELEC ’90., 12th International*, pp. 543–550, Oct. 1990.
- [35] Y. Wen, R. Wolski, and C. Krintz, “Online Prediction of Battery Lifetime for Embedded and Mobile Devices,” in *IN PROCEEDINGS OF PACS*, p. 2004, 2003.
- [36] W. Ross and P. Budney, “Development of a battery runtime prediction algorithm and a method for determining its accuracy,” in *Telecommunications Energy Conference, 1995. INTELEC ’95., 17th International*, pp. 277–283, 1995.
- [37] L. Benini, A. Macii, E. Macii, and M. Poncino, “Discharge current steering for battery lifetime optimization,” in *Proceedings of the 2002 inter-*

national symposium on Low power electronics and design, ISLPED '02, (New York, NY, USA), pp. 118–123, ACM, 2002.

- [38] V. Pop, H. J. Bergveld, P. H. L. Notten, J. H. G. O. het Veld, and P. P. L. Regtien, “Accuracy analysis of the State-of-Charge and remaining runtime determination for lithium-ion batteries,” *Measurement*, vol. 42, no. 8, pp. 1131–1138, 2009.
- [39] B. Saha and K. Goebel, “Battery Data Set,” tech. rep., NASA Ames, Moffett Field, CA, 2007.
- [40] C. Laugier and R. Chatila, *Autonomous Navigation in Dynamic Environments*. Springer, 1 ed., 2007.
- [41] Y. Chen, B. Long, and X. Lei, “The Battery State of Charge Estimation Based Weighted Least Squares Support Vector Machine,” in *Power and Energy Engineering Conference (APPEEC), 2011 Asia-Pacific*, pp. 1–4, Mar. 2011.
- [42] J. Lu, S. Liu, Q. Wu, and Q. Qiu, “Accurate modeling and prediction of energy availability in energy harvesting real-time embedded systems,” in *Green Computing Conference, 2010 International*, pp. 469–476, 2010.
- [43] B. Akdemir, S. Gunes, and I. T. Comlekçiler, “Microcontroller Compatible Sealed Lead Acid Battery Remaining Energy Prediction Using Adaptive Neural Fuzzy Inference System,” in *Web Information Systems*

- and Mining, 2009. WISM 2009. International Conference on*, pp. 779–783, 2009.
- [44] D. Di Domenico, G. Fiengo, and A. Stefanopoulou, “Lithium-ion battery state of charge estimation with a Kalman Filter based on a electrochemical model,” in *Control Applications, 2008. CCA 2008. IEEE International Conference on*, pp. 702–707, Ieee, 2008.
- [45] P. S. Maybeck, *Stochastic Models, Estimation and Control*. New York: Academic Press, 1 ed., 1979.
- [46] D. J. Higham., “An Algorithmic Introduction to Numerical Simulation of Stochastic Differential Equations,” *SIAM Rev.*, vol. 43, pp. 525–546, Mar. 2001.
- [47] N. Kazantzis, K. T. Chong, J. H. Park, and A. G. Parlos, “Control-Relevant Discretization of Nonlinear Systems With Time-Delay Using Taylor-Lie Series,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 127, no. 1, p. 153, 2005.
- [48] A. Gelb, *Applied optimal estimation*. MIT Press, 1974.
- [49] J. M. Aughenbaugh and C. J. J. Paredis, “The Value of Using Imprecise Probabilities in Engineering Design,” *Journal of Mechanical Design*, vol. 128, no. 4, pp. 969–979, 2006.
- [50] E. A. Wan and R. Van Der Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Adaptive Systems for Signal Processing, Com-*

- munications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pp. 153–158, 2000.
- [51] S. J. Julier, Jeffrey, and K. Uhlmann, “Unscented Filtering and Nonlinear Estimation,” in *Proceedings of the IEEE*, pp. 401–422, 2004.
- [52] N. Metropolis and S. M. Ulam, “The Monte Carlo Method,” *Journal of the American Statistical Association*, vol. 44, no. 247, pp. 335–341, 1949.
- [53] S. Thrun, “Particle Filters in Robotics,” in *in Proceedings of the 17th Annual Conference on Uncertainty in AI (UAI)*, 2002.
- [54] A. Doucet and N. Gordon, “Particle filters for state estimation of jump Markov linear systems,” *Signal Processing, IEEE*, vol. 49, no. 3, pp. 613–624, 2001.
- [55] S. Butler and J. Ringwood, “Particle filters for remaining useful life estimation of abatement equipment used in semiconductor manufacturing,” in *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on DOI - 10.1109/SYSTOL.2010.5675984*, pp. 436–441, 2010.
- [56] K. Arras, “An introduction to error propagation: Derivation, meaning and examples,” tech. rep., Autonomous Systems Lab, Institute of Robotic Systems, Swiss Federal Institute of Technology Lausanne (EPFL), 1998.
- [57] Gelb A. and W. E. V. Velde, *Multiple-Input Describing Functions and Nonlinear System Design*. McGraw Hill, 1968.

- [58] S. J. Julier, “The Scaled Unscented Transformation,” pp. 4555–4559, 2002.
- [59] M. Briers, S. R. Maskell, and R. Wright, “A rao-blackwellised unscented Kalman filter,” in *Information Fusion, 2003. Proceedings of the Sixth International Conference of*, vol. 1, pp. 55–61, 2003.
- [60] S. Yang, D. Wen, J. Sun, and J. Ma, “Gaussian sum particle filter for spacecraft attitude estimation,” in *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, vol. 3, pp. V3–566 –V3–570, July 2010.
- [61] G. Terejanu, “Unscented Kalman filter tutorial,” Tech. Rep. 1, 2011.
- [62] L. Tang, J. DeCastro, G. Kacprzyński, K. Goebel, and G. Vachtsevanos, “Filtering and prediction techniques for model-based prognosis and uncertainty management,” in *Prognostics and Health Management Conference, 2010. PHM '10.*, pp. 1–10, 2010.
- [63] J. Zhou, D. Liu, Y. Peng, and X. Peng, “Dynamic battery remaining useful life estimation: An on-line data-driven approach,” in *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, pp. 0–3, 2012.
- [64] J. V. Candy, *Bayesian signal processing: classical, modern, and particle filtering methods*. John Wiley and Sons, 2009.

- [65] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *Signal Processing, IEEE Transactions on*, vol. 50, pp. 174–188, Feb. 2002.
- [66] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, “Metrics for evaluating performance of prognostic techniques,” in *Prognostics and Health Management, 2008. PHM 2008. International Conference on*, pp. 1–17, 2008.
- [67] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, “Evaluating prognostics performance for algorithms incorporating uncertainty estimates,” *2010 IEEE Aerospace Conference*, pp. 1–11, Mar. 2010.
- [68] L. Tang and M. Orchard, “Novel metrics and methodologies for the verification and validation of prognostic algorithms,” in *Aerospace Conference, 2011 IEEE*, pp. 1–8, 2011.
- [69] N. Lybeck, S. Marble, and B. Morton, “Validating Prognostic Algorithms: A Case Study Using Comprehensive Bearing Fault Data,” *2007 IEEE Aerospace Conference*, pp. 1–9, 2007.
- [70] H. Yu, F. Tseng, and R. McGee, “Driving pattern identification for EV range estimation,” *2012 IEEE International Electric Vehicle Conference*, pp. 1–7, Mar. 2012.

- [71] D. N. Rakhmatov and S. B. K. Vrudhula, “An analytical high-level battery model for use in energy management of portable electronic systems,” in *Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design, ICCAD '01*, (Piscataway, NJ, USA), pp. 488–493, IEEE Press, 2001.
- [72] M. Chen and G. A. Rincon-Mora, “Accurate electrical battery model capable of predicting runtime and I-V performance,” *Energy Conversion, IEEE Transactions on*, vol. 21, pp. 504–511, June 2006.
- [73] D. Panigrahi, C. Chiasserini, S. Dey, R. Rao, K. Lahiri, and E. al., “Battery Life Estimation of Mobile Embedded Systems,” in *IN PROC. INT. CONF. VLSI DESIGN*, pp. 55–63, 2001.
- [74] M. R. Jongerden and B. R. Haverkort, “Which battery model to use?,” 2009.
- [75] J. Wong, *Theory of Ground Vehicles*. John Wiley and Sons, 2001.
- [76] S. Pandit and S.-M. Wu, *Time Series and System Analysis With Applications*. Krieger Publishing Company, 2001.
- [77] H. Hahn, S. Meyer-Nieberg, and S. Pickl, “Electric load forecasting methods: Tools for decision making,” *European Journal Of Operational Research*, vol. 199, no. 3, pp. 902–907, 2009.

- [78] E. A. Feinberg and D. Genethliou, "Load Forecasting," in *Applied Mathematics for Restructured Electric Power Systems*, pp. 269–285, Springer, 2005.
- [79] Y. Hida, R. Yokoyama, K. Iba, K. Tanaka, and K. Yabe, "Load forecasting on demand side by multi-regression model for operation of battery energy storage system," in *Universities Power Engineering Conference (UPEC), 2009 Proceedings of the 44th International*, pp. 1–5, 2009.
- [80] D. G. Lee, B. W. Lee, and S. H. Chang, "Genetic programming model for long-term forecasting of electric power demand," *Electric Power Systems Research*, vol. 40, pp. 17–22, 1997.
- [81] Y.-S. Lee and L.-I. Tong, "Forecasting energy consumption using a grey model improved by incorporating genetic programming," *Energy Conversion and Management*, vol. 52, pp. 147–152, Jan. 2011.
- [82] C. A. Brooks and K. Iagnemma, "Self-Supervised Terrain Classification for Planetary Surface Exploration Rovers," *Journal of Field Robotics*, pp. 1–24, 2012.
- [83] S. A. Pourmousavi Kani and G. H. Riahy, "A new ANN-based methodology for very short-term wind speed prediction using Markov chain approach," *2008 IEEE Electrical Power and Energy Conference Energy Innovation*, pp. 1–6, 2008.

- [84] R. Mamlook and O. Badram, “A fuzzy inference model for short-term load forecasting,” *Energy Policy*, vol. 4, no. 37, pp. 1239–1248, 2009.
- [85] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*. Springer, 2001.
- [86] J. M. Lujano-Rojas, R. Dufo-López, and J. L. Bernal-Agustín, “Optimal sizing of small wind/battery systems considering the DC bus voltage stability effect on energy capture, wind speed variability, and load uncertainty,” *Applied Energy*, vol. 93, pp. 404–412, May 2012.
- [87] D. C. Montgomery, C. L. Jennings, and M. Kulahci, *Introduction to Time Series Analysis and Forecasting*. Wiley, 1 ed., 2008.
- [88] S. Soliman and A. Al-Kandari, *Electrical Load Forecasting: Modeling and Model Construction*. Butterworth-Heinemann, 1 ed., 2010.
- [89] R. Matias and A. Carvalho, “Comparison analysis of statistical control charts for quality monitoring of network traffic forecasts,” *Systems, Man, and . . .*, pp. 404–409, 2011.
- [90] A. Bard and L. Faulkner, *Electrochemical Methods: Fundamentals and Applications*. New York, NY, USA: John Wiley and Sons, Inc., 2nd ed., 2001.
- [91] R. A. Huggins, *Energy Storage*. Springer, 2010.

- [92] R. E. Garcia, Y.-M. Chiang, W. Craig Carter, P. Limthongkul, and C. M. Bishop, “Microstructural Modeling and Design of Rechargeable Lithium-Ion Batteries,” *Journal of The Electrochemical Society*, vol. 152, no. 1, p. A255, 2005.
- [93] M. Doyle, J. P. Meyers, and J. Newman, “Computer Simulations of the Impedance Response of Lithium Rechargeable Batteries,” *Journal of the Electrochemical Society*, vol. 147, no. 1, p. 99, 2000.
- [94] M. D. Levi and D. Aurbach, “Simultaneous Measurements and Modeling of the Electrochemical Impedance and the Cyclic Voltammetric Characteristics of Graphite Electrodes Doped with Lithium,” *The Journal of Physical Chemistry B*, vol. 101, no. 23, pp. 4630–4640, 1997.
- [95] H. He, R. Xiong, and J. Fan, “Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach,” *Energies*, vol. 4, pp. 582–598, Mar. 2011.
- [96] D. Rakhmatov and S. B. K. Vrudhula, “Time-to-failure estimation for batteries in portable electronic systems,” in *Low Power Electronics and Design, International Symposium on, 2001.*, pp. 88–91.
- [97] C. Chiasserini and R. Rao, “Energy efficient battery management,” *Selected Areas in Communications, . . .*, vol. 19, no. 7, pp. 1235–1245, 2001.
- [98] C. Chiasserini and R. Rao, “A traffic control scheme to optimize the battery pulsed discharge,” *Military Communications Conference . . .*, 1999.

- [99] R. Peng and M. Pedram, "Battery-aware power management based on Markovian decision processes," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 25, no. 7, pp. 1337–1349, 2006.
- [100] P. Spagnol, S. Rossi, and S. M. Savaresi, "Kalman Filter SoC estimation for Li-Ion batteries," in *Control Applications (CCA), 2011 IEEE International Conference on*, pp. 587–592.
- [101] B. Saha, K. Goebel, S. Poll, and J. Christophersen, "An integrated approach to battery health monitoring using bayesian regression and state estimation," in *Autotestcon, 2007 IEEE*, pp. 646–653, 2007.
- [102] M. Yang, Y. Liu, and Z. You, "The Reliability of Travel Time Forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 162–171, Mar. 2010.
- [103] S. B. Stancliff, J. M. Dolan, A. Trebi-Ollennu, R. S. J. I. Conference, I. Robots, and S. October, "Mission Reliability Estimation for Multi-robot Team Design," in *Robotics Institute*, (Beiging, China), pp. 2206–2211, 2006.
- [104] R. Remenyte-Prescott, J. Andrews, and P. Chung, "An efficient phased mission reliability analysis for autonomous vehicles," *Reliability Engineering & System Safety*, vol. 95, pp. 226–235, Mar. 2010.

- [105] J. Andrews, J. Poole, and W. Chen, “Fast mission reliability prediction for Unmanned Aerial Vehicles,” *Reliability Engineering & System Safety*, pp. 1–7, Mar. 2013.
- [106] P. Tompkins, A. Stentz, and D. Wettergreen, “Mission-level path planning and re-planning for rover exploration,” *Robotics and Autonomous Systems*, vol. 54, pp. 174–183, Feb. 2006.
- [107] G. Kewlani, G. Ishigami, and K. Iagnemma, “Stochastic mobility-based path planning in uncertain environments,” *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1183–1189, Oct. 2009.
- [108] H. Wei, B. Wang, Y. Wang, Z. Shao, and K. C. Chan, “Staying-alive path planning with energy optimization for mobile robots,” *Expert Systems with Applications*, vol. 39, pp. 3559–3571, Feb. 2012.
- [109] S. Lu, H. Lu, and W. Kolarik, “Multivariate performance reliability prediction in real-time,” *Reliability Engineering & System Safety*, vol. 72, pp. 39–45, Apr. 2001.
- [110] Z. Xu, Y. Ji, D. Zhou, and S. Member, “A New Real-Time Reliability Prediction Method for Dynamic Systems Based on On-Line Fault Prediction,” *IEEE Transactions on Reliability*, vol. 58, no. 3, pp. 523–538, 2009.

- [111] J. Li, D. W. Coit, and E. a. Elsayed, “Reliability modeling of a series system with correlated or dependent component degradation processes,” *2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pp. 388–393, June 2011.
- [112] M. Luo, D. Wang, M. Pham, C. Low, J. Zhang, D. Zhang, and Y. Zhao, “Model-based fault diagnosis/prognosis for wheeled mobile robots: a review,” *31st Annual Conference of IEEE Industrial Electronics Society, 2005. IECON 2005.*, p. 6 pp., 2005.
- [113] S. Narasimhan, E. Balaban, M. Daigle, I. Roychoudhury, A. Sweet, J. Celaya, and K. Goebel, “Autonomous Decision Making for Planetary Rovers Using Diagnostic and Prognostic Information,” in *8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, (Mexico City), 2012.
- [114] D. Liu, J. Pang, J. Zhou, Y. Peng, and M. Pecht, “Prognostics for state of health estimation of lithium-ion batteries based on combination Gaussian process functional regression,” *Microelectronics Reliability*, vol. 53, pp. 832–839, June 2013.
- [115] D. Tilbury and A. Ulsoy, “Reliable operations of unmanned ground vehicles: Research at the Ground Robotics Reliability Center,” in *IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*, 2010.

- [116] C. C. Ooi and C. Schindelbauer, “Minimal Energy Path Planning for Wireless Robots,” *Mobile Networks and Applications*, vol. 14, pp. 309–321, Jan. 2009.
- [117] T. Wang and C. Cassandras, “Optimal control of batteries with fully and partially available rechargeability,” *Automatica*, vol. 48, no. 8, pp. 1658–1666, 2012.
- [118] F. Dressler and G. Fuchs, “Energy-aware operation and task allocation of autonomous robots,” *Proceedings of the Fifth International Workshop on Robot Motion and Control, 2005. RoMoCo '05.*, pp. 163–168, 2005.
- [119] S. Qin and T. a. Badgwell, “A survey of industrial model predictive control technology,” *Control Engineering Practice*, vol. 11, pp. 733–764, July 2003.
- [120] T. Geyer, G. Papafotiou, and M. Morari, “Model Predictive Control in Power Electronics: A Hybrid Systems Approach,” *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 5606–5611, 2005.
- [121] M. Pedram and W. Qing, “Design considerations for battery-powered electronics,” in *Design Automation Conference, 1999. Proceedings. 36th*, pp. 861–866.
- [122] H. Kim, K. G. Shin, and K. Hahnsang, “Scheduling of Battery Charge, Discharge, and Rest,” in *Real-Time Systems Symposium, 2009, RTSS 2009. 30th IEEE*, pp. 13–22, 2009.

- [123] R. Rao and S. Vrudhula, “Battery optimization vs energy optimization: which to choose and when?,” in *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, pp. 439–445.
- [124] R. Meyer, R. A. Decarlo, P. H. Meckl, C. Doktorcik, and S. Pekarek, “Hybrid Model Predictive Power Flow Control of a Fuel Cell-Battery Vehicle,” in *American Controls Conference*, pp. 2725–2731, 2011.
- [125] S. Teleke and M. Baran, “Optimal control of battery energy storage for wind farm dispatching,” in *IEEE Transactions on Energy Conversion*, vol. 25, pp. 787–794, 2010.
- [126] G. Klančar and I. Škrjanc, “Tracking-error model-based predictive control for mobile robots in real time,” *Robotics and Autonomous Systems*, vol. 55, pp. 460–469, June 2007.
- [127] Y. Mei, Y. Lu, and Y. Hu, “A case study of mobile robot’s energy consumption and conservation techniques,” *Advanced Robotics*, pp. 492–497, 2005.
- [128] R. Cuzner and G. Venkataramanan, “The status of DC micro-grid protection,” *Applications Society Annual*, pp. 1–8, 2008.
- [129] J. Carlson and R. Murphy, “Reliability analysis of mobile robots,” *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 1, pp. 274–281.

- [130] T. Kim and P. J.S., “The Current Move of Lithium Ion Batteries Towards the Next Phase,” *Advanced Energy Materials*, vol. 2, no. 7, pp. 860–872, 2012.
- [131] J. Gertler, “U.S. Unmanned Aerial Systems,” tech. rep., Congressional Research Service, 2012.
- [132] M. Orchard, M. Cerda, B. Olivares, and J. Silva, “Sequential Monte Carlo methods for Discharge Time Prognosis in Lithium-Ion Batteries,” *International Journal of Prognostics and Health Management*, pp. 1–12, 2012.
- [133] J. Thangavelautham, “Lithium hydride powered PEM fuel cells for long-duration small mobile robotic missions,” in *Robotics and Automation*, (Saint Paul, MN), pp. 415–422, IEEE, May 2012.
- [134] S. Thomas, “How green are electric vehicles?,” *International Journal of Hydrogen Energy*, vol. 37, no. 7, pp. 6053–6062, 2012.
- [135] M. A. Gross and G. Cardell, “An overview of nasas mars science laboratory,” in *European Space Power Conference ESPC*, no. 1, (Pasadena, CA), 2011.
- [136] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, “Discrete-Time Markov Jump Linear Systems,” 2005.

- [137] T. Qian Ying and C. Spanos, “Non-Gaussian uncertainty propagation in statistical circuit simulation,” in *Quality Electronic Design (ISQED), 2011 12th International Symposium on*, pp. 1–8.
- [138] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, vol. 39, no. 1, pp. 1–38, 1977.
- [139] C. Bouman, M. Shapiro, G. Cook, C. Atkins, and H. Cheng, “Cluster: An unsupervised algorithm for modeling Gaussian mixtures,” tech. rep., 1997.
- [140] H. Akaike, “A new look at the statistical model identification,” *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.
- [141] M. Daigle, B. Saha, and K. Goebel, “A comparison of filter-based approaches for model-based prognostics,” in *Aerospace Conference, 2012* . . . , pp. 1–10, Ieee, Mar. 2012.
- [142] E. Zio and G. Pelsoni, “Particle filtering prognostic estimation of the remaining useful life of nonlinear components,” *Reliability Engineering & System Safety*, vol. 96, pp. 403–409, Mar. 2011.
- [143] A. C. Rencher and W. F. Christensen, *Methods of Multivariate Analysis*. Wiley, 2002.

- [144] D. Karnopp, D. Margolis, and R. Rosenberg, *System Dynamics: Modeling, Simulation, and Control of Mechatronic Systems*. John Wiley and Sons, Inc., 4th ed., 2006.
- [145] J. Kassakian, M. Schlecht, and G. Verghese, *Principles of power electronics*. Reading Massachusetts: Addison-Wesley Publishing, 1991.
- [146] J. R. Lesage and R. G. Longoria, “Modeling and Synthesis Methods for Retrofit Design of Submarine Actuation Systems,” tech. rep., Office of Naval Research, Arlington, VA, 2011.
- [147] F. Daum, “Nonlinear filters: beyond the Kalman filter,” *Aerospace and Electronic Systems Magazine, IEEE*, no. August, pp. 57–69, 2005.
- [148] Y. Suhov and M. Kelbert, *Markov Chains, Primer in Random Processes and their Applications*. Cambridge University Press, 2008.
- [149] M. Ceraolo and G. Pede, “Techniques for estimating the residual range of an electric vehicle,” *IEEE Transactions on Vehicular Technology*, vol. 50, no. 1, pp. 109–115, 2001.
- [150] M. Buet, J. Pearson, and D. S. Bennett, “Robotic Mining System for Rapid Earth Orbit Capture of Asteroid Resources Simplicity and highest possible space-proven reliability are emphasized in designing and engineering the overall AMS design .,” tech. rep., Georgia Institute of Technology, Atlanta, GA, 2013.

- [151] J. Carlson, R. Murphy, and a. Nelson, “Follow-up analysis of mobile robot failures,” *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pp. 4987–4994 Vol.5, 2004.
- [152] F. Kendoul, “Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems,” *Journal of Field Robotics*, vol. 29, no. 2, pp. 315–378, 2012.
- [153] W. Y. Lee, H. Kim, H. Lee, L. Wan Yeon, K. Hyogon, and L. Heejo, “Maximum-Utility Scheduling of Operation Modes With Probabilistic Task Execution Times Under Energy Constraints,” *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 28, no. 10, pp. 1531–1544, 2009.
- [154] H. Chen and H. a. Rakha, “Prediction of dynamic freeway travel times based on vehicle trajectory construction,” *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 576–581, Sept. 2012.
- [155] C.-H. Wu, J.-M. Ho, and D. Lee, “Travel-Time Prediction With Support Vector Regression,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, pp. 276–281, Dec. 2004.
- [156] N. Chowdhury and C. Leung, “Improved travel time prediction algorithms for intelligent transportation systems,” *Knowledge-Based and Intelligent Information*, pp. 355–365, 2011.

- [157] T. Idé and S. Kato, “Travel-time prediction using Gaussian process regression: A trajectory-based approach,” in *Proc. of the 9th SIAM international conference on Data Mining*, pp. 1185–1196, 2009.
- [158] I. Skog, P. Händel, and S. Member, “In-Car Positioning and Navigation Technologies A Survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp. 4–21, 2009.
- [159] Y. Wang and S. Boyd, “Fast Model Predictive Control Using Online Optimization,” *IEEE Conference on*, vol. 18, no. 2, pp. 267–278, 2010.
- [160] W. Kwon and S. Han, *Receding horizon control: model predictive control for state models*. London: Springer, 2005.
- [161] J. Maciejowski, *Predictive control: with constraints*. New York, NY, USA: Prentice Hall, 2002.
- [162] A. Bemporad and M. Morari, “Robust Model Predictive Control : A Survey,” *Robustness in Identification and Control*, vol. 245, 1999.
- [163] A. S. Dutka, A. W. Ordys, and M. J. Grimble, “Nonlinear Predictive Control of 2 dof helicopter model,” in *Conference on Decision and Control*, no. December, (Maui, Hawaii), pp. 3954–3959, IEEE, 2003.
- [164] P. Falcone, M. Tufó, F. Borrelli, J. Asgari, and H. E. Tseng, “A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems,” *2007 46th IEEE Conference on Decision and Control*, pp. 2980–2985, 2007.

- [165] R. Fletcher, *Practical Methods of Optimization*. Wiley, 2000.
- [166] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, 2000.
- [167] J.-J. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1991.
- [168] R. Bartlett, A. Wachter, and L. Biegler, “Active set vs. interior point strategies for model predictive control,” in *American Control Conference*, no. June, 2000.
- [169] P. Mauracher and E. Karden, “Dynamic modeling of lead/acid batteries using impedance spectroscopy for parameter identification,” *Journal of Power Sources*, vol. 67, pp. 69–84, 1997.
- [170] L. P. Mandal and R. W. Cox, “A transient-based approach for estimating the electrical parameters of a lithium-ion battery model,” in *Energy Conversion Congress and Exposition (ECCE), 2011 IEEE*, pp. 2635–2640, 2011.
- [171] M. Thele, O. Bohlen, D. U. Sauer, and E. Karden, “Development of a voltage-behavior model for NiMH batteries using an impedance-based modeling concept,” *Journal of Power Sources*, vol. 175, pp. 635–643, Jan. 2008.
- [172] S. Shaw and C. Laughman, “A Method for Nonlinear Least Squares With Structured Residuals,” *IEEE Transactions on Automatic Control*, vol. 51, pp. 1704–1708, Oct. 2006.

- [173] T. Hu, B. Zanchi, and J. Zhao, “Determining battery parameters by simple algebraic method,” in *American Control Conference*, pp. 1–6, 2011.
- [174] D. Andrea, *Battery Management Systems for Large Lithium Ion Battery Packs*. Artech House, 1st ed., 2010.
- [175] S. Piller, M. Perrin, and A. Jossen, “Methods for state-of-charge determination and their applications,” *Journal of Power Sources*, vol. 96, no. 1, pp. 113–120, 2001.
- [176] Q. Miao, L. Xie, H. Cui, W. Liang, and M. Pecht, “Remaining useful life prediction of lithium-ion battery with unscented particle filter technique,” *Microelectronics Reliability*, vol. 53, pp. 805–810, Dec. 2012.
- [177] J. R. Lesage and R. G. Longoria, “Hybrid Observer Design for Online Battery State-of-Charge Estimation,” in *American Control Conference*, (Washington, DC, USA), IEEE, 2013.
- [178] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, 2006.

Vita

Jonathan Robert LeSage was born in Temple, Texas on December 4, 1985, the son of Gene and Jannette LeSage. Jonathan completed his undergraduate education at The University of Texas at Austin in 2008 with a Bachelors of Science in Mechanical Engineering. Additionally, Jonathan worked as a industry researcher in both Houston, Texas and Clamart, France in 2007 and 2008 respectively.

Returning to academics after his industry experience, Jonathan enrolled in the Dynamic Systems and Controls area of the Department of Mechanical Engineering at The University of Texas at Austin in the fall of 2008. Upon entering the program, he was hired as a graduate research assistant at the Applied Research Laboratories under the supervision of Bill Shutt and the academic guidance of Dr. Raul Longoria.

Permanent address: 2836 San Gabriel St
Austin, Texas 78705

This dissertation was typeset with \LaTeX^\dagger by the author.

[†] \LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's \TeX Program.