This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

Fast Model Predictive Control for Redistributive Lithium Ion Battery Balancing

Lucas McCurlie, *Student Member*, *IEEE*, Matthias Preindl, *Member*, *IEEE*, and Ali Emadi, *Fellow*, *IEEE*

***Abstract*—Energy storage systems with Lithium Ion batteries require balancing due to individual cells having manufacturing inconsistencies, different self discharge rates, internal resistances and temperature variations. Non-dissipative redistributive balancing further improves on the pack capacity and efficiency over a Dissipative approach where energy is consumed across shunt resistors. This paper presents a high level, fast model predictive control in continuous time. The optimization problem uses performance metrics to balance the SoC in the battery pack.It is shown in simulation that MPC achieves a single point convergence of the state of charge when compared against a common rule based algorithm. This improves the efficiency of the power electronics and prolongs the life of each battery cell since frequent switching between charging and discharging of intermediate cells is avoided. Experimental results are presented to show a Redistributive battery balancing system that achieves a balanced state in the minimum amount of time by coupling the Fast MPC with microcontrollers available on todays market.**

***Index Terms*—Control strategy, electric vehicle, fast model predictive control, hybrid vehicle, Lithium-ion battery, redistributive cell balancing**

I. Introduction

ELECTRIC Vehicles (EV) have gained significant attention due to elevated atmospheric pollution, a spreading concern for the reliance on fossil fuels as well as harsher government policies on carbon emissions and greenhouse gases. However, wide adoption of EVs requires improvements in battery technology [1]. Many applications such as aircraft e-taxis, hybrid diesel trains, electrified buses and electric vehicles use Lithium Ion batteries because of their high energy density, low self-discharge rates and high cell voltage. When constructing a Energy Storage System (ESS) for these applications, many cells are connected in series to achieve higher power requirements, but will lead to a nearly exponential reduction in the battery life as the number of cells increases [2]. This reduction of battery life is primarily caused by cell imbalances over time which occur due to manufacturing inconsistencies, different self discharge rates, internal resistances and temperature variations.

Protection of Lithium-ion batteries experiencing over and under voltages is essential to maximizing the health and safety of the cells [3]. If over voltage occurs, production of CO_2, C_2H_4 and other gases will increase the internal temperature and pressure causing severe battery damage or an explosion [4]. If under voltage occurs, internal reactions cause the cell to

Manuscript received January 21, 2016; revised June 4, 2016; accepted August 11, 2016. This work was supported in part by the Canada Excellence Research Chairs Program.

L. McCurlie and A. Emadi are with MacAUTO, McMaster University, Hamilton, ON L8P0A6, Canada (e-mail: mccurllb@mcmaster.ca; emadi@mcmaster.ca)

M. Preindl is with the Department of Electrical Engineering, Columbia University in the City of New York, 1300 S.W. Mudd MC 4712, 500 W. 120th Street, New York NY 10027 (phone: +1-212-853-0791; e-mail: matthias.preindl@columbia.edu)

TABLE I
BALANCING SYSTEM PARAMETERS

Lithium Ion Battery	Symbol	Value/Unit
Number of cells	n	6
Number of links	m	6
Battery Cell number	j	1 to n
Link number	l	1 to m
Rated Cell Capacity	$\widehat{C}$	8 Ah
Normalized Balancing current	$\bar{u}$	$\lvert\bar{u}_z\rvert \leq 1$
Measured Cell Voltage	V_c	2.5 to 4.2 V
Stack Voltage	V_s	$\sum_{j=1}^{n}(V_{cj})$ V
Nominal Cell Voltage	$\widehat{V}_n$	3.7 V
Average cell current	$\bar{i}_c$	A
Average stack current	$\bar{i}_\psi$	A
Flyback Converter	**Symbol**	**Value/Unit**
Cell side inductance	L_c	3.05 μH
P.E Switching Period	T_p	0.5μs - 15μs
Turns Ratio	N_r	1:2
Converter efficiency	η	90%
Maximum Power	$\widehat{P}$	2W - 2.5W
Primary peak current	$\widehat{I}_p$	10A
Secondary peak current	$\widehat{I}_s$	5A
Instantaneous Primary current	i_p	A
Instantaneous Secondary current	i_s	A
General System	**Symbol**	**Value/Unit**
High level Sample Time	T_s	180 s
Operational Sample Time	T_a	30 s
Balance Time	T_b	$\leq T_a$ s
Maximum Link Current	M_a	3.5 A
Maximum Operational Link Current	$\widehat{I}_L$	0.517 A
Estimated Discharge Current	$\widetilde{i}_d$	A
Estimated Charge Current	$\widetilde{i}_c$	A

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

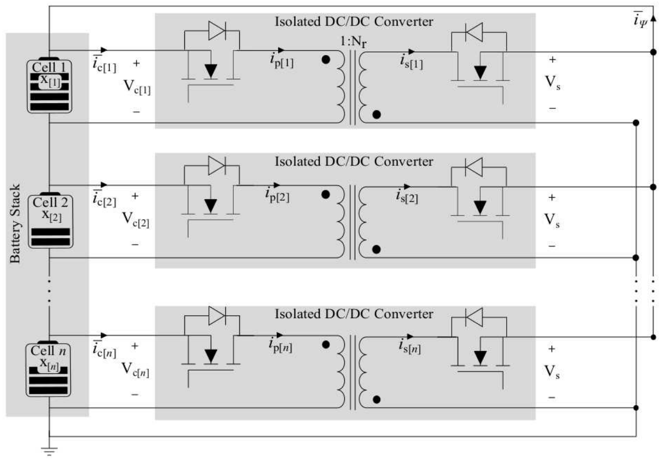

Fig. 1. Individual Cell to Stack topology showcasing battery string, power electronics and connections.

lose a large part of its capacity. The voltage of a battery cell is related to its remaining energy i.e state of charge (SoC). Therefore, a weak cell is defined as one that has a lower SoC than the others in the pack. Likewise a strong cell is one that has a higher SoC. Without a on-board balancing system, the cells capacities would drift apart causing weak cells to dominate discharging time and strong cells to dominate charging time. A Battery Management System (BMS) is implemented to avoid the harmful effects of cell imbalances, improve the effective capacity of the pack and keep each cell within a predefined operational safety region.

A balancing system is categorized as either Dissipative or Non-Dissipative. The Dissipative balancing approach draws excess energy from strong cells then dissipate this energy as heat through external shunt resistors [5], [6]. This method, although inexpensive is wasteful of energy that can be re-purposed elsewhere. The Non-dissipative redistributive technique shuttles the excess energy from the strong cells into the weak ones using power electronics [7]–[9]. Three approaches of achieving redistributive balancing are cell to cell (C2C), cell to stack (C2S) and stack to cell (S2C). A C2C approach transfers the excess energy between adjoining cells. A C2S approach transfers the excess energy from strong cells then redistributes it back onto the battery stack. Likewise, a S2C approach transfers the excess energy from the battery stack to the weak cells. By Combining the last two methods, it is possible to simultaneously charge and discharge individual cells.

This paper shows how to model the system by taking advantage of slow time varying cell dynamics and average currents through the power electronics. A fast Model Predictive Control (MPC) balancing algorithm is then proposed based on the performance metrics outlined in [9]. In order to solve the MPC algorithm, a simple Linear Programming (LP) solver is developed for Microcontrollers available on todays market. Simulation results are presented which compare a Rule based strategy (RBS) to the MPC approach. The RBS method has many switching periods of intermediate cells that hinder the efficiency of the power electronics and reduce each cells battery life in the system. The implementation details are provided by means of a low level actuation strategy coupled with the fast MPC to apply the correct amount of balancing current. Finally, experimental results are shown for balancing a redistributive C2S (discharge) and S2C (charge) configuration based on a multiple transformer topology that uses bi-directional flyback converters to realize the battery currents. The nomenclature used in this paper is shown in TABLE I.

II. Battery System Description

A battery pack in a redistributive balancing system is defined by n series connected battery cells with m number of links. Each cell is described by the amount of charge via $\mathbf{Q}_x x(t) \in \mathbb{R}^n$. The matrix

$$\mathbf{Q}_x = \begin{bmatrix} \widehat{C}_1 & 0 & \dots & 0 \\ 0 & \widehat{C}_2 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & \widehat{C}_n \end{bmatrix} \in \mathbb{R}^{nxn}$$

is a diagonal matrix defining charge capacities for cell 1 to n respectively and the state of charge (SoC) is

$$x(t) = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^\top \in \mathbb{R}^n.$$

Each element in the SoC $x(t)$ vector ranges between zero and one where the value of 0 corresponds to a completely empty cell and the value of 1 corresponds to a fully charged cell. For the system to become balanced, charge is moved between m links. The balancing current being transfered through the links is $\mathbf{Q}_u u(t) \in \mathbb{R}^m$ where $u(t)$ is a vector containing the normalized balancing currents and

$$\mathbf{Q}_u = \begin{bmatrix} \widehat{I}_{L1} & 0 & \dots & 0 \\ 0 & \widehat{I}_{L2} & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & \widehat{I}_{Lm} \end{bmatrix} \in \mathbb{R}^{m\times m}$$

defines a diagonal matrix containing the maximum current each link can handle.

The connection between n cells and m links is defined by a topology matrix $\mathbf{T} \in \mathbb{R}^{nxm}$. It describes how the balancing charge is transfered (from and to each cell). The bi-directional C2S/S2C configuration based on a multiple transformer topology from [9] and shown in Fig. 1 is used in this research where

$$\mathbf{T} = \begin{bmatrix} \frac{1}{n} - 1 & \frac{1}{n} & \dots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} - 1 & \dots & \frac{1}{n} \\ \vdots & \vdots & & \vdots \\ \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} - 1 \end{bmatrix} \in \mathbb{R}^{n\times m}.$$

This topology is used for reference where the concepts can be applied to other topologies found in [9]. In the multiple transformer topology, charge is removed from one cell and

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

distributed equally amongst all the cells in the stack. Likewise, the charge can be removed from the entire stack then added to a single cell. Each cell has its own unique link to the stack which allows for simultaneous movement of charge to and from multiple cells.

The charge stored in a battery cell is modeled using simple continuous time integrator dynamics as

$$\mathbf{Q}_x \dot{x}(t) = \mathbf{T}\mathbf{Q}_u u(t), \tag{1}$$

where the topology matrix $\mathbf{T}$ relates the normalized balancing currents $u(t)$ with the SoC $x(t)$. A sign convention of $u(t) > 0$ indicates a flow of charge from the cell to the stack and $u(t) < 0$ indicates a flow of charge from the stack the cell. The system dynamics can now be simplified to

$$\dot{x}(t) = \mathbf{B}u(t), \tag{2}$$

where $\mathbf{B} = \mathbf{Q}_x^{-1}\mathbf{T}\mathbf{Q}_u$. Thus, a battery cells state of charge in terms of a control input at the final balancing time τ, is

$$x(\tau) = x(0) + \mathbf{B}\int_0^\tau u(t)dt, \tag{3}$$

where τ is the time to balance, i.e. the time required to balance the battery pack. The maximum rated link current limits the applied balancing current. These balancing currents are subject to polyhedral constraints that depend on the topology [9] [10]. We define a inequality constraint based on the maximum amount of current through each link $|u_l| \leq 1$ for $l = 1, 2, ..., m$. Thus the inequality constraint set for this topology is

$$u(t) \in \{u \in \mathbb{R}^m | -1 \leq u_l \leq 1\} \tag{4}$$

From **Proposition 1** [9] there exists a constant input trajectory $u(t) = \bar{u}$ such that

$$x(\tau) = x(0) + \mathbf{B}\bar{u}\tau. \tag{5}$$

The equality constraint on the system dynamics is defined by transforming (5) into a regulation problem using the transformation matrix

$$\mathbf{L} = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n}.$$

This matrix $\mathbf{L}$ removes the average SoC and leaves the unbalanced SoC such that the equality constraint now becomes

$$\mathbf{L}x(0) + \mathbf{L}\mathbf{B}\bar{u}\tau = \mathbf{L}x(\tau) = 0. \tag{6}$$

The constraints for the system are written as $\mathbf{H}\bar{u} \leq K$ for the inequality constraint and $\mathbf{H}_{eq}\bar{u} = K_{eq}$ for the equality constraint.

III. Predictive Control

In [9] performance metrics have been developed to evaluate hardware. However, this paper proposes to use the minimum time to balance (t2b) metric as a balancing control. This metric is used for reference and the concepts can be extended for minimum energy to balance (e2b) or a combination of both. In the previous section, a constant balancing current trajectory $\bar{u}$ must be found such that the battery cells become balanced in a finite amount of time τ. This constant current will ensure a single point of convergence of the SoC. This reduces any micro-cyles between charging and discharging of the intermediate battery cells [11] and increases the balancing efficiency.

The constrained linear optimization problem is formally expressed as

$$\tau^\star(x) = \underset{\tau \geq 0}{\text{minimize}}\, \tau \tag{7a}$$

$$\text{subject to } \mathbf{L}x + \mathbf{L}\mathbf{B}\bar{u}\tau = 0 \tag{7b}$$

$$\mathbf{H}\bar{u}\tau - \mathbf{K}\tau \leq 0 \tag{7c}$$

To efficiently solve (7a) using popular Linear programming solver packages such as LPSOLVE, CPLEX or MOSEK, it is worth reproducing the problem in standard form. This is achieved by defining a new column vector containing both variables

$$z = \begin{bmatrix} v \\ \tau \end{bmatrix},$$

where $v = \bar{u}\tau$. It is important to realize that $\mathbf{L}x$ is a parameter and is treated as a constant. The minimum time to balance problem (7a) in standard form becomes

$$z^\star = \text{minimize}\, g'z \tag{8a}$$

$$\text{subject to } \mathbf{A}_{eq}z = b_{eq} \tag{8b}$$

$$\mathbf{A}z \leq b \tag{8c}$$

where $\mathbf{A}_{eq} = [LB\ 0]$, $b_{eq} = [-\mathbf{L}x\ 0]$, $\mathbf{A} = [\mathbf{H} - k], b = 0$.

Many simple solvers are more efficient when dealing without equality constraints such as qpOASES. This is not mandatory but by removing the equality constraint from (8a), the dimensions of the optimization problem are reduced.

To do this, we introduce a linear transformation

$$z = \mathbf{F}\bar{z} + z_0 \tag{9}$$

where $\mathbf{F}$ is the nullspace of $\mathbf{A}_{eq}$, such that $\mathbf{F}\mathbf{A}_{eq} \equiv 0$ and z_0 is any solution of $\mathbf{A}_{eq}z_0 = b_{eq}$. Equation (9) is substituted into (8a) to ensure that the equality constraint is still satisfied but is now removed from the problem. Thus, we formally define the minimum time to balance problem in standard form with the removed equality constraint as

$$\bar{z}^\star = \text{minimize}\, g'\bar{z} \tag{10a}$$

$$\text{subject to } \bar{\mathbf{A}}\bar{z} \leq \bar{b} \tag{10b}$$

where $\bar{\mathbf{A}} = \mathbf{A}\mathbf{F}$, $\bar{b} = b - \mathbf{A}z_o$.

The performance metric is now used to define a model predictive controller. The controller is executed in discrete time steps k and at each time step the optimal control problem is solved. The problem is reiterated using the updated state. Hence, we introduce the discrete time dynamics

$$\mathbf{L}x + \mathbf{L}\mathbf{B}\bar{u}\tau = 0. \tag{11}$$

To adapt this into a discrete time MPC controller, at each sampling time instant kT_s, the problem requires information regarding the SoC of each battery module in the system. The

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

state of charge x is not measurable but can be estimated according to $y = C(x)$ where y is the voltage associated with the battery terminals and C is some nonlinear mapping [12]. Advanced methods for SoC estimation are Kalman filters [13], neural networks [14], electrochemical impedance spectroscopy [15] and fuzzy logic [16], [17]. For the development of the fast MPC controller, we assume the SoC is reconstructed with sufficient precision. By solving (10a), we then find the optimal τ^* and $\bar{u}^*$. We introduce a scaling factor ν that scales down the control if τ^* is smaller than the sampling period T_s such that the process leads to the following closed loop dynamics

$$x[k+1] = x[k] + \nu T_s \mathbf{B} \bar{u}^*[k] \tag{12}$$

where

$$\nu = \begin{cases} \frac{\tau^*}{T_s} & \text{for } \tau^* \leq T_s \\ 1 & \text{for } \tau^* > T_s \end{cases} \tag{13}$$

This sequence is repeated for each sampling instant until the system becomes balanced, i.e $\mathbf{L}x[k] = 0$.

IV. Solver

Several commercial and open-source software packages are available to solve linear programming (LP) problems. Some packages allow easy integration with MATLAB and Simulink as well as other simulation software. The inputs of the solver are typically the parameter matrices that define the cost function and constraints in standard form. If the problem is feasible, the solver produces a suitable approximation of the optimal vector. However, applying LP solvers on available microcontrollers (MCU) is challenging because (1) the code needs to be recompiled for a given platform that rules out commercial software where the source code is not available; (2) MCU's support almost exclusively c-programming language that rules out large solver projects that have cross-dependencies on other programming languages or are optimized for x86 architectures with assembly code or processor specific extensions; (3) the RAM of MCU's is typically measured in kB (68kB for the widely used TI F28335, 512kB for the TI F28377D that is used in this work) and even compact LP software, e.g. LPSolve, require several MB of program memory. Regarding (3), RAM can certainly be added to a control board but this approach increases complexity and cost. In this research, a "micro (μ) solver" is implemented to show the viability of the proposed fast MPC control method.

TABLE II
SOLVING TIME COMPARISON RESULTS

Dimensions	6	12	18	24	32	40	80
μ-solver (sec)	0.124	0.345	0.338	0.366	0.372	0.386	4.211
qpOASES (sec)	0.014	0.085	0.189	0.222	0.278	0.325	0.462
CPLEX (sec)	0.216	0.215	0.218	0.219	0.216	0.217	0.274

TABLE III
SOLVING SOLUTION COMPARISON RESULTS

	qpOASES (Benchmark)	μ-solver (Simulation)	μ-solver (Experimental DSP)
$\bar{u}[k]$	0.747968	0.74797	0.74797
	0.166862	0.166862	0.16686
	0.404288	0.404289	0.40429
	0.252569	0.252569	0.25257
	-1.00000	-1.00000	-1.0010
	1.00000	1.00000	1.0010
τ(min)	20.982	20.9994	20.9993

A simple solver method that can operate with limited memory is the gradient method. This approach is typically not used as an LP solver due to efficiency concerns. The projected gradient method has linear time complexity [18] that is (on average) slower compared to other solvers, e.g. the simplex method or an interior point method. However, battery balancing can be implemented with relatively large sampling periods such that solver time is less critical compared to memory usage. The gradient method uses the gradient of the cost function (10a) to find the optimum iteratively. The method starts with an initial guess $z_i = z_0$ at iteration $i = 0$. The direction of steepest descent i.e the search direction

$$\delta z = -\nabla_z((10\text{a}) : g'z) = -g, \tag{14}$$

is constant for an LP because the minimum is always located in a vertex of the hyperplane. For an LP, the cost function does not define the minimum. It is found by iteratively using a max decent step and a projected step. Thus, the optimal vector is updated according to

$$y_{i+1} = z_i + \delta z, \tag{15}$$

where z_i is the guess for an optimum at iteration i. The vector y_{i+1} does not necessarily satisfy the constraints. Hence it is projected onto the feasible set $\mathcal{C} = \{z \in \mathbb{R}^n \mid \mathbf{A}z \leq b\}$, i.e.

$$z_{i+1} = \text{proj}_{\mathcal{C}} y_{i+1}. \tag{16}$$

Where $\text{proj}_{\mathcal{C}}$ denotes the orthogonal projection onto the affine $\mathcal{C}$. In practice, the projection is obtained iteratively with a Van Neumann type algorithm. The decent step will eventually violate the constraints and the projected step ensures we obtain a feasible point. It is noted that also this projection method has a linear time complexity and is used due to simplicity. The resulting solver has been compared to CPLEX and qpOASES in TABLE II. The results show that the μ-solver can solve small problems reasonably fast. However, the μ-solver performs suboptimal at higher dimensions, i.e. large battery stacks (due to the projection operation) and will be subject to further research. Evaluating the results, it should be further taken into account that the μ-solver is run as interpreted MATLAB code (ported via code generation onto the MCU platform) and CPLEX is highly optimized for x86 platforms and called through the MATLAB interface.

To reduce the solver time, the gradient algorithm is combined with two techniques. The first is a warm start scheme that uses an initial guess to reduce the number of iterations

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

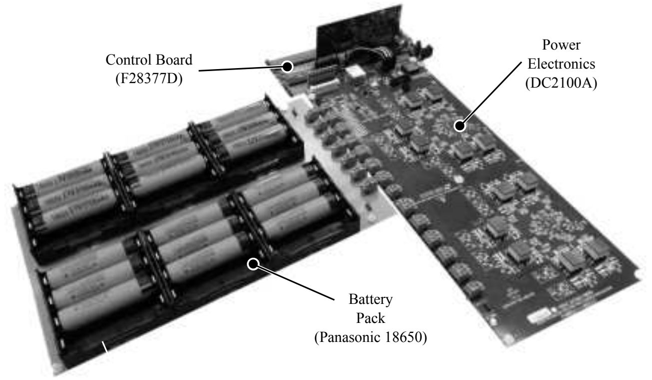

Fig. 2. Physical test bench with battery pack, balancing hardware, monitoring and control board. The Lithium Ion battery pack is comprised of 3 Panasonic 18650 cells in parallel (module) and 6 modules in series.

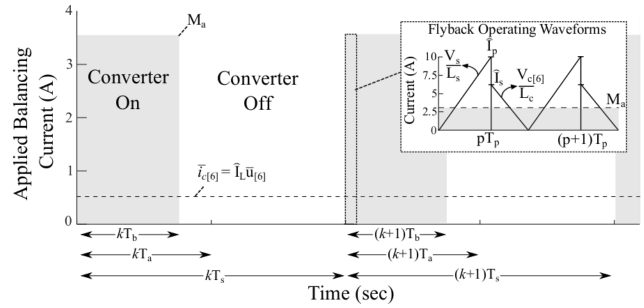

Fig. 3. Low Level, low frequency actuation scheme illustrating a Constant Operating Point Modulation (COPM) strategy with high frequency flyback operating waveforms

required to converge onto the solution. Since MPC requires SoC to solve a similar optimization problems multiple times, the solution of the $k-1$ sampling instant can be used as initial guess at time step k. The second technique is early termination that stops the solver after a predetermined number of iterations. Although this may result in a sub-optimal control sequence, it is guaranteed to be feasible due to the projection operation [19]. In practice, these techniques often provide good performance since the measurements tend to be similar in two adjacent sampling instants [20]. To further evaluate the μ-solver, TABLE III shows a comparison with qpOASES in a simulated environment and on the experimental DSP. The optimal balancing currents $\bar{u}$ and the time to balance τ are solved at the first sampling time instant i.e $k=1$. The results show a nearly identical solution in all three environments.

V. Implementation Details

A. Test Bench

To showcase the non-dissipative redistributive balancing approach, experiments are conducted on the testbench shown in Fig. 2. The first main component in the system is a modified DC2100A demo board from Linear Technology. On this board is a LTC-6804 monitoring chip that measures the cells voltages. It has internal over/under voltage protection and conveys the information via SPI to the control DSP. It also features a LTC-3300 chip that controls the mosfets for the flybacks converters. Each flyback module operates in critical mode utilizing a pulse frequency modulation (PFM) strategy descried in [21]. How this works is when the primary switch for a module is closed while discharging a cell, it measures the instantaneous primary current i_p until a maximum peak current $\widehat{I}_p$ is reached. It then opens the primary switch and closes the secondary switch to allow the instantaneous secondary current i_s to be released back onto the stack. The flybacks are bi-directional meaning the process is mirrored for when a battery cell is charging. Each converter module operates with the waveforms shown in Fig. 3 - Flyback Operating Waveforms.

The second main component is a custom built battery pack using Panasonic NCR 18650 cells. In order to handle high current transfer that the DC2100A outputs, the battery pack consists of 6 modules connected in series. Each module has 3 Lithium Ion battery cells connected in parallel. The last component in the system is a Texas Instruments F28377D dual core micro controller (DSP). The primary function of core one is to solve for the optimal normalized balancing current based on the cells state of charge. The solver described in section IV is flashed onto the first core of the DSP. It is first written in MATLAB as functions blocks, then code compiled into C files using MATLAB Coder. This method allows for fast prototyping and comparison between a simulated environment and what is on the DSP. The second core handles all Battery Management type activities and is connected through SPI to the DC2100A demo board. Its main tasks are to read cell voltage, determine state of charge, control how long each mosfet is on/off for, and to send information to core 1.

B. Low level Control

To actuate the desired current from the fast MPC controller, a upper level, low frequency control strategy referred to as constant operating point modulation (COPM) [21] must be utilized. The fast MPC solves for a normalized current $\bar{u}$ that when multiplied with the upper bound on actuating current through the power electronics $\widehat{I}_L$, is how much current the controller thinks the batteries receives i.e $\bar{i}_c = \widehat{I}_L\bar{u}$. In reality, the hardware used to test is either on or off, and operating with much higher current M_a. The relationship between the two currents $\widehat{I}_L$ and M_a is determined offline as

$$\widehat{I}_L = \frac{M_a T_a}{T_s}. \tag{17}$$

At each kT_s time step shown in Fig. 3, we translate $\bar{i}_c = \widehat{I}_L\bar{u}$ into an effective current that works with the COPM strategy. In [21], estimated current equations provide a close approximation of the average link current through each flyback converter. While a cell is being discharged, the estimated current through the link is

$$\widetilde{i}_d = \frac{\widehat{I}_p V_s}{2(V_s + N_r V_c)}. \tag{18}$$

The values of the estimated currents are then scaled with the maximum operating link current M_a. These calculations are performed by the DSP in order to control how long each switch is on for to achieve the same effective current the fast MPC controller requires.

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

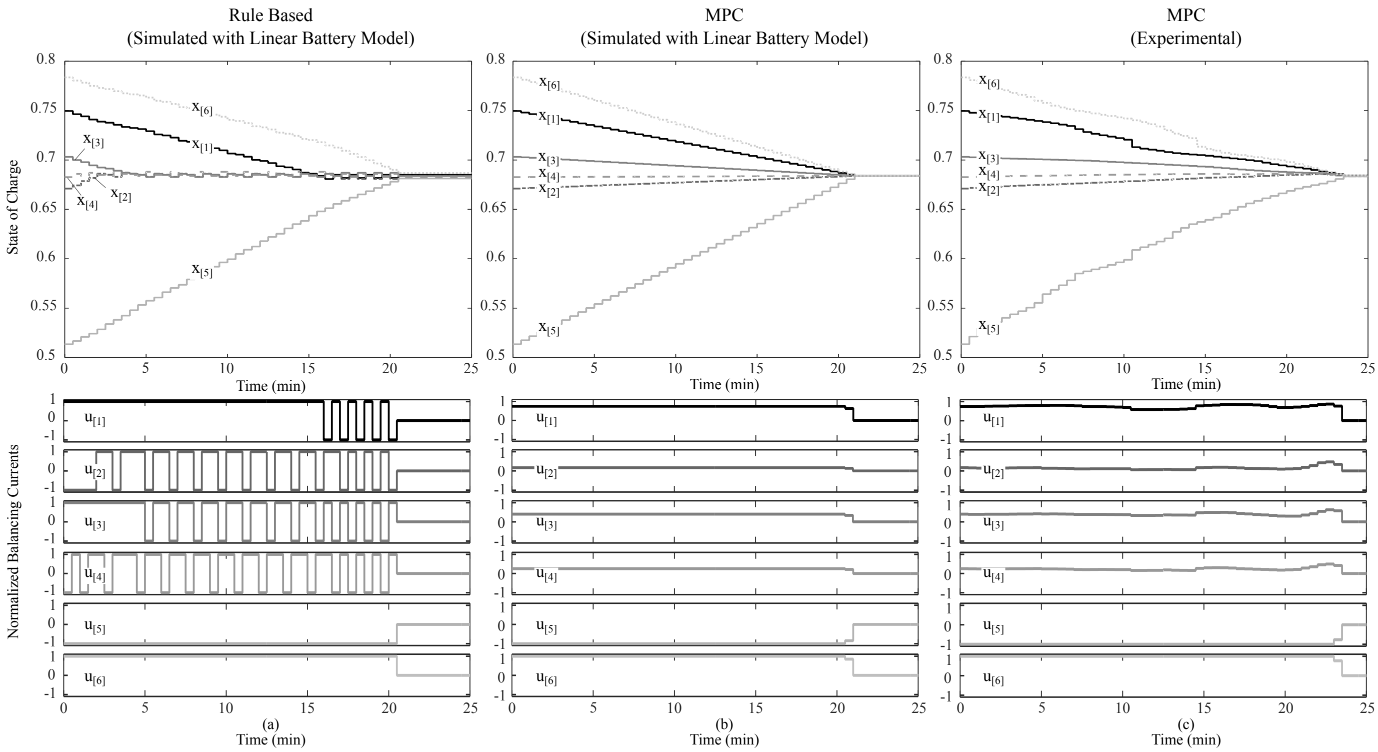

Fig. 4. Balancing results for (a) Rule based strategy in simulation using a Linear Battery Model, (b) MPC in simulation using a Linear Battery Model, (c) MPC on experimental test setup. The initial SoC values are 0.749, 0.671, 0.703, 0.682, 0.513, 0.783 for cell 1 through 6 respectively.

VI. Results

A. Simulation - RBS Vs. MPC

In this section, the rule based strategy (RBS) for battery balancing control is defined then compared against the MPC algorithm. The working principles of a RBS approach is to compare each cells SoC with the average SoC. If a cell has a higher SoC than the average, it is discharged onto the stack. Likewise, if a cell has a lower SoC than the average, it is charged by the stack. This simple control method uses the maximum link current available i.e each cell is always charging or discharging until all cells SoC fall within a "balanced zone". An arbitrary balanced zone is defined in this paper as 5% and will ensure the balancing stops. Achieving the single point of convergence in charge levels is unobtainable due to the current flowing in each cell is always the maximum. The intermediate cells will reach the average faster than the maximum and minimum cells. Operating under a RBS control on a simulated environment using a linear battery model, the state of charge and normalized balancing currents are shown in Fig. 4a.

The following statements are made when compared with the SoC and balancing currents for the MPC approach in Fig. 4b. During RBS control, cells 2-4 converge faster than cells 1,5 and 6. The main issue with this is that high switching currents hinder the efficiency of the power electronics as more conduction loss and switching loss occur through the mosfets. The normalized balancing currents from Fig. 4a show that a reduction in the expected life of the Lithium-ion batteries will occur due to high C-rates and large number of charge-discharge cycles [11].

The power consumed by the non-dissipative redistributive battery balancing system is now compared when using the two control strategies, RBS and MPC. Before the test begins, the amount of unbalanced energy is calculated as

$$E_u = \sum_{j=1}^{n} V_n \widehat{C} |\mathbf{L}x[j]|. \tag{19}$$

This defines how much energy needs to be shuttled around in order to achieve a balanced state. Then during each sampling period T_s, the amount of consumed power is defined as

$$E_L = \sum_{i=1}^{m} V_n \widehat{I}_L |\bar{u}[i]|(1-\eta). \tag{20}$$

Fig. 5c shows that the consumed power is doubled when using RBS, thus is less efficient. This is because the converters are always on i.e cells are always being charged or discharged. This shows a clear motivation behind adopting the MPC approach which is to apply more of a constant current that results in a single point of convergence between the SoC values.

B. Experimental

In this section, we simulate a closed loop system and compare the results with that of an experimental one. We evaluate the controller and balancing hardware for a system containing 6 cells in series, each with a rated capacity of $\widehat{C} = 8Ah$ and each link with a maximum of $\widehat{I}_L = 0.52A$. The balancing hardware is carried out using flyback converters between each cell connecting to the stack of batteries in a

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

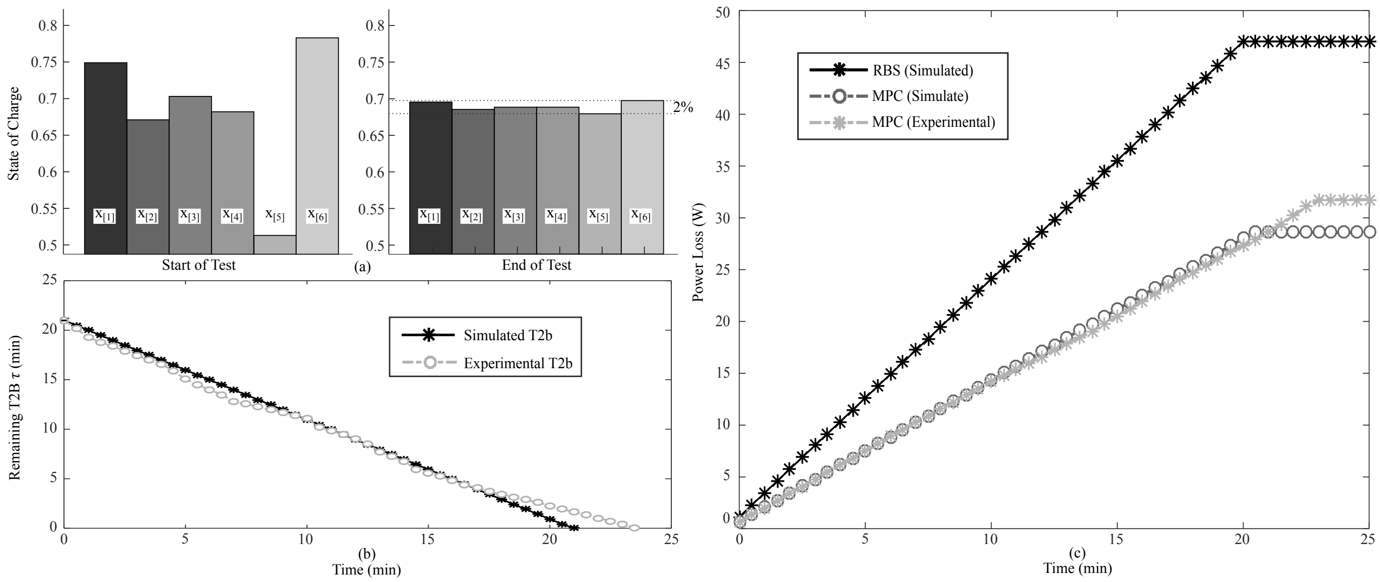

Fig. 5. Test Results (a) Start of test SoC Vs. End of test SoC, (b) time to balance results using MPC Simulation vs Experimental; (c) comparing Power Consumption in Watts during operating time for RBS and MPC

individual cell to stack topology. The fast MPC controller decides a reference current based on the cells state of charge values then the low level control actuates that desired amount of current for each kT_s period.

We performed an experimental test for a system that has an initial unbalanced state with a 27% difference between the highest and lowest SoCs. The experimental state of charge and normalized balancing currents are shown in fig 2. These results are compared to that of the simulated ones shown in fig 3. The balancing currents remain constant over the majority of the operating window. Slight deviations in a constant current can be observed but the reason for this difference comes from the SoC construction. In simulation a Linear battery model is used to obtain SoC. However, on the experimental setup, SoC is more difficult to estimate with such high precision, which is not the focus of this paper. When comparing these results to RBS control for the same initial unbalance, an improvement in the battery life and balancing efficiency is achieved due to less switching between charging and discharging cycles and a single point of converge in SoC. More over, we compute the minimum time to balance at each kT_s time step during the experiment. The experimental time to balance τ deviates slightly from the simulated τ. Again, this is most likely caused by imperfections of the estimation of state of charge. However, the results shown in Fig. 5b remain practically on target to that of the simulated controllers and thus shows that the the TI F28377D dual core controller is able to consistently solve for the optimal balancing current over the entire test window.

According to [22], a specification of balanced is defined when all battery cells SoC is within a 3% margin. We define a more aggressive margin set to a 2% difference between all SoC values. Fig. 5a shows the start of test SoCs with an initial unbalanced of 27% difference. After balancing has occurred, the experimental system stops when the target margin of 2% is reached and a 20mV difference in voltage is measured.

VII. Conclusions

This paper showcases experimental results for a fast MPC control approach to balance the cells state of charge inside of a battery pack. The fast MPC controller may be adapted for different topologies found in [9]. The algorithms presented here within are adoptable for any series/parallel cell configuration. This allows for mismatched cell capacities to be balanced. The minimum-time optimization problem is known to be a bounded LP (convex), where the projected gradient method is known to find the optimum in a finite number of steps. The gradient method was used for simplicity and the ability to operate with a small memory footprint (necessary for the microcontroller implementation). Existing solvers use libraries (e.g. for linear algebra) that cannot easily be compiled for a specific microcontroller platform. The authors show that the proposed gradient method works reasonably well for small problems. An improved μ-solver scalability is currently being investigated.

A system using 6 cells in series with an initial difference of 27% SoC is conducted to gain the experimental results. The fast MPC controller is developed based on the minimum time to balance performance metric in [9]. The fast MPC controller applies the maximum input $\bar{u}$ if $\tau > T_s$. If $\tau \leq T_s$ then the input is scaled by a $\frac{\tau}{T_s}$ factor. The MPC approach is compared to a rule based strategy and shows that constant current will result in a single point of converge of the SoCs. This reduces any micro-cyles between charging and discharging of the intermediate battery cells [11] and improves the balancing efficiency. With the use of bi-directional flyback converters, a redistributive non-dissipative battery balancing topology reaches a balanced state in minimum time.

VIII. Acknowledgments

This research was undertaken, in part, thanks to funding from the Canada Excellence Research Chairs Program.

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2611488, IEEE Transactions on Industrial Electronics

References

[1] S. Abu-Sharkh and D. Doerffel, “Rapid test and non-linear model characterisation of solid-state lithium-ion batteries,” *J. Power Sources*, vol. 130, no. 1-2, pp. 266–274, Feb 2004.

[2] N. Kutkut, H. Wiegman, D. Divan, and D. Novotny, “Charge equalization for an electric vehicle battery system,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 1, Aug 1998.

[3] M. Isaacson, R. Hollandsworth, P. Giampaoli, F. Linkowsky, A. Salim, and V. Teofil, “Advanced Lithium Ion Battery Charger,” in *Battery Conf. on Applic. and Advances*, Aug 2000.

[4] A. Affanni, A. Bellini, G. Franceschini, P. Guglielmi, and C. Tassoni, “Battery choice and management for new-generation electric vehicles,” *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1343–1349, Sept 2005.

[5] B. Lindemark, “Individual cell voltage equalizers (ICE) for reliable battery performance,” in *Inter. Telecomm. Energy Conf.*, vol. 91, August 1991.

[6] C. Moo, Y. Hsieh, and I. Tsai, “Charge equalization for series-connected batteries,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 2, pp. 704–710, Sept 2003.

[7] N. Kutkut and D. Divan, “Dynamic equalization techniques for series battery stacks,” in *Inter. Telecomm. Energy Conf.*, July 1996.

[8] J. Cao, N. Schofield, and A. Emadi, “Battery balancing methods: A comprehensive review,” in *IEEE Veh. Power and Prop. Conf.*, Aug 2008.

[9] M. Preindl, C. Danielson, and F. Borrelli, “Performance evaluation of battery balancing hardware,” in *Euro. Control Conf.*, July 2013.

[10] M. Caspar and S. Hohmann, “Optimal Cell Balancing with Model-based Cascade Control by Duty Cycle Adaption,” *Inter. Fed. of Auto. Control*, pp. 10 311–10 318, Aug 2014.

[11] E. Chemali, L. McCurlie, B. Howey, T. Stiene, M. M. Rahman, M. Preindl, R. Ahmed, and A. Emadi, “Minimizing Battery Wear in a Battery-Ultracapacitor Hybrid Energy Storage System using a Linear Quadratic Regulator,” in *IEEE Conf. of Indus. Elect. Society*, Jan 2015.

[12] B. Xiao and Y. Shi, “A universal state-of-charge algorithm for batteries,” in *Design Automation Conf.*, July 2010.

[13] G. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. Part 1: Background,” *J. Power Sources*, vol. 161, no. 2, pp. 1356–1368, July 2004.

[14] M. Charkhgard and M. Farrokhi, “State-of-charge estimation for lithium-ion batteries using neural networks and EKF,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 12, pp. 4178–4187, Feb 2010.

[15] D. Howey, P. Mitcheson, V. Yufit, G. Offer, and N. Brandon, “Online measurement of battery impedance using motor controller excitation,” *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2557–2566, July 2014.

[16] R. Feng, S. Zhao, and X. Lu, “On-line estimation of Dynamic State-of-Charge for lead acid battery based on fuzzy logic,” in *Inter. Conf. on Meas., Info. and Control*, no. 1, March 2013.

[17] R. Ling, Y. Dong, H. Yan, M. Wu, and Y. Chai, “Fuzzy-PI control battery equalization for series connected lithium-ion battery strings,” in *IEEE Power Electr. and Motion Control Conf.*, vol. 4, no. 5, Oct 2012.

[18] A. Domahidi, A. Zgraggen, M. Zeilinger, M. Morari, and C. Jones, “Efficient Interior Point Methods for Multistage Problems Arising in Receding Horizon Control,” in *IEEE Conf. on Decisions and Control*, Feb 2012.

[19] M. Preindl, S. Bolognani, and C. Danielson, “Model Predictive Torque Control with PWM using fast gradient method,” in *IEEE Appl. Power Electr. Conf. and Expo.*, July 2013.

[20] Y. Wang and S. Boyd, “Fast Model Predictive Control Using Online Optimization,” *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 6974–6979, June 2010.

[21] L. McCurlie, M. Preindl, P. Malysz, and A. Emadi, “Simplified Control for Redistributive Balancing Systems using Bidirectional Flyback Converters,” in *IEEE Transport. Electri. Conf. and Expo.*, July 2015.

[22] A. Imtiaz and F. Khan, “Time shared flyback converter based regenerative cell balancing technique for series connected li-ion battery strings,” *IEEE Trans. Power Electron.*, vol. 28, no. 12, pp. 5960–5975, July 2013.

Lucas McCurlie (S’15) is a research engineer at McMaster Institute for Automotive Research and Technology. He received the B.Tech degree in Process Automation (summa cum laude) and M.A.Sc degree in electrical and computer engineering from McMaster University, Hamilton, ON, Canada in 2013 and 2016 respectively. His main research interests include battery management systems and optimal control strategies.

Matthias Preindl (GSM’12 - M’14) is an Assistant Professor in the Department of Electrical Engineering, Columbia University in the City of New York, USA. He received the PhD degree in Energy Engineering from the Doctoral School of Industrial Engineering at University of Padua, Italy (2014), the MSc degree in Electrical Engineering and Information Technology from ETH Zurich, Switzerland (2010), and the BSc degree in Electrical Engineering (summa cum laude) University of Padua, Italy (2008). M. Preindl was a Sessional Professor in the Department of Electrical and Computer Engineering (2015) and a Post Doctoral Research Associate at the McMaster Institute for Automotive Research and Technology (2014-2015), both at McMaster University, Canada. M. Preindl leads the Motor Drives and Power Electronics Laboratory at Columbia University. His primary research interests are design and control of power electronic and motor drive systems with applications in electrified transportation systems and smart grids.

Ali Emadi (S’98-M’00-SM’03-F’13) received the B.S. and M.S. degrees in electrical engineering with highest distinction from Sharif University of Technology, Tehran, Iran, in 1995 and 1997, respectively, and the Ph.D. degree in electrical engineering from Texas A&M University, College Station, TX, USA, in 2000. He is the Canada Excellence Research Chair in Hybrid Powertrain at McMaster University in Hamilton, Ontario, Canada. Before joining McMaster University, Dr. Emadi was the Harris Perlstein Endowed Chair Professor of Engineering and Director of the Electric Power and Power Electronics Center and Grainger Laboratories at Illinois Institute of Technology (IIT) Chicago, Illinois, USA. He is the principal author/coauthor of over 350 journal and conference papers as well as several books including Vehicular Electric Power Systems (2003), Energy Efficient Electric Motors (2004), Uninterruptible Power Supplies and Integrated Power Electronic Converters and Digital Control (2009).

0278-0046 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.