Active Cell Balancing by Model Predictive Control for Real Time Range Extension

Jun Chen, *Senior Member, IEEE*, Aman Behal and Chong Li, *Senior Member, IEEE*

***Abstract*— This paper studies the active cell balancing problem by using model predictive control (MPC) for real time range extension. Specifically, three MPC formulations are proposed and compared: the first one being a tracking controller to force all cells to follow the same trajectory generated by a nominal cell model, the second one trying to maximize the lowest cell SOC/voltage and the last one minimizing the difference between the highest and lowest cell SOC/voltages. Both steady state and transient conditions are simulated to assess the effectiveness of the proposed controllers, and a range extension of 4% is found for dynamic driving cycle and 7% for steady state condition. Comparing to the literature, our approaches achieve similar range extension, without making the restrictive assumption that the final battery state-of-charge is known in advance, making our approaches more applicable. Real time implementability is demonstrated via throughput analysis.**

I. Introduction

State-of-charge (SOC) and voltage imbalance commonly exist in battery cell, partially due to manufacturing variation, and such imbalance inevitably degrades the battery performance and reduces the range of electric vehicle (EV) [1]–[3]. The cell imbalance issue can be mitigated by cell balancing circuit, such as flyback DC/DC converter [2] and half-bridge converter [4], especially under conditions of higher power demand and high variation [5]. Active cell balancing methods can be either *dissipative* or *nondissipative*, [2], [6]. In this paper, we focus on nondissipative cell balancing control, which has less energy waste for performing balancing.

Active cell balancing control has been studied in the literature. In [7] a rule-based control strategy was adopted for cell balancing, where both voltage imbalance and SOC imbalance were considered in the criterion to trigger control action. Reference [8] studied cell balancing problem in the context of renewable energy integration in the power grid, where heterogeneous battery systems with different types, ages, and rated capacity, were interconnected. A simple feedback controller was utilized in [8] to calculate the balancing current.

Advanced control methods, such as model predictive control (MPC), have also been investigated in the literature for active cell balancing. In [6], an auxiliary power module was designed to perform balancing during charging by using MPC with linear SOC model. Linear MPC was also adopted in [9] to track a reference trajectory generated by assuming the final SOC was known in advance. Though such assumption was very restrictive, a 5% of range increase was shown through simulation. The authors of [9] also demonstrated the robustness against the unknown driving cycle. Reference [1] considered both SOC and voltage dynamics of battery cells and formulated the balancing control problem as a reachability analysis problem, where benefit on range extension was shown on a short driving cycle. Reference [10] utilized nonlinear MPC to simultaneously minimize SOC imbalance and energy waste through balancing current, with simulation results on a two-cell battery. To fit into a microcontroller, [11] considered fast MPC where the objective is to minimize the time to balance with a linear dynamic model for balancing current.

In this paper, we study MPC for active cell balancing problem for EV range extension, and investigate three balancing objectives for MPC. In the first objective, MPC is set to track the SOC/voltages of all cells to follow a short term reference that is generated by using a nominal battery cell model. This setup is similar to the MPC formulation of [9], without assuming that the final SOC at the end of the drive cycle is known in advance. In the second objective, MPC is set to maximize the lowest cell SOC/voltage, as opposed to track all cells' SOC/voltages. In the last objective, MPC is set to minimize the difference between the highest and lowest cell SOC/voltages. Note that the goal of active cell balancing is to push all cell's voltages away from the lower bound, below which a cell would fail and lead to the failure of the entire battery pack. The three MPC formulations presented above realize this goal by using different cost functions, whose effectiveness will be investigated. Numerical simulation of these three MPCs are presented and it was found that the first formulation is favored for transient conditions while the last two formulations are comparably favored for steady state conditions. A 4% range extension is shown through simulation over dynamic driving cycle (e.g., Federal Test Procedure [FTP]) and 7% for steady state condition. Finally, real time implementability is demonstrated by throughput analysis.

The rest of this paper is organized as follows. Section II presents the equivalent circuit model for each cell and the whole battery pack, while Section III formulates the optimal control problems and three MPC setups. Section IV presents numerical simulation results, and the paper is concluded in Section V.

This work was supported in part by the faculty startup fund from School of Engineering and Computer Science at Oakland University.

Jun Chen is with the Department of Electrical and Computer Engineering, Oakland University, Rochester, MI 48309, USA (email: junchen@oakland.edu).

Aman Behal is with the Electrical and Computer Engineering and NanoScience Technology Center, University of Central Florida, Orlando, FL 32816, USA (e-mail: abehal@ucf.edu).

Chong Li is with the Department of Electrical Engineering, Columbia University, NYC, 10027, USA (email: cl3607@columbia.edu).

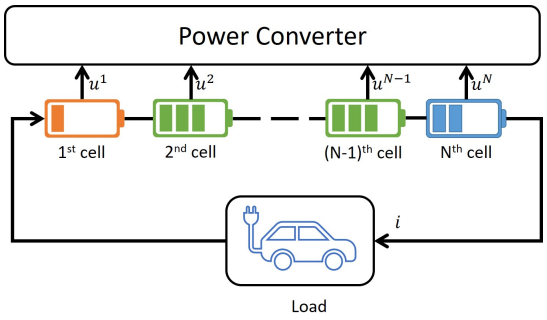

Fig. 1. Structure of series connected battery cell and balancing current.

II. Battery Model

The system considered here is shown in Fig. 1, where N cells are connected in series to provide the current i requested from the higher level controller. Due to manufacturing variation, the SOC and voltage of each cell can be significantly different without balancing mechanism. The cell balancer, in this case a power converter circuit, draws current u^n from an arbitrary cell n and transport it to another cell j, so that the variations of SOC and voltage among the series connected cells are minimized. Note that we only consider series connection here, while the proposed control methodology can be readily applied to the case where parallel connection are also present.

To model the dynamics of each cell and the overall battery, we use equivalent circuit model (ECM), which has been widely used in the literature to study the dynamic behavior of Li-Ion battery [12]–[16]. We briefly introduce the model used in this study as follows, and more details can be found from the aforementioned references.

The dynamics are specified by

$$\dot{s^n} = -\eta^n \frac{i^n}{C^n} \tag{1a}$$

$$\dot{V_p^n} = -\frac{V_p^n}{R_p^n C_p^n} + \frac{i^n}{C_p^n} \tag{1b}$$

$$y^n = V_{oc}^n - V_p^n - i^n R_o^n, \tag{1c}$$

where the superscript n denotes the nth cell, s^n is the cell SOC, η^n is the coulombic efficiency of cell n, C^n is the cell capacity in Amp Hour, V_p^n is the relaxation voltage over R_p^n, V_{oc}^n is the open circuit voltage, y^n is the terminal voltage, and i^n is the battery pack current. We use the convention that positive value of i^n indicates discharging from battery and negative indicates charging. Note that V_{oc}^n, R_o^n, R_p^n, and C_p^n are all dependent of s^n, making (1) a nonlinear model. Refer to [15] for an example of such dependency for a nominal cell.

Equation (1) can be discretized using Euler's method, with sampling time T_s, as follows

$$s_{k+1}^n = s_k^n - \eta^n \frac{T_s}{C^n} i_k^n \tag{2a}$$

$$V_{p,k+1}^n = V_{p,k}^n - \frac{T_s}{R_p^n C_p^n} V_{p,k+1}^n + \frac{T_s}{C_p^n} i_k^n \tag{2b}$$

$$y_k^n = V_{oc,k}^n - V_{p,k}^n - i_k^n R_o^n. \tag{2c}$$

Denote $\zeta^n := [s^n, V_p]^T$ where $\cdot^T$ denotes matrix/vector transpose, then one can write

$$\zeta_{k+1}^n = f^n(\zeta_k^n, i_k + u_k^n) \tag{3a}$$

$$y_k^n = g^n(\zeta_k^n, i_k + u_k^n). \tag{3b}$$

Note that here we use the fact that for the battery with structure in Fig. 1, the current i_k^n drawn through cell n equals the pack current i_k plus balancing current u_k^n. Define $\zeta = [\zeta^1, \zeta^2, \dots, \zeta^N]^T$ as the state vector for the entire battery pack and y to be the terminal voltage of the battery pack, then

$$\zeta_{k+1} = \begin{bmatrix} f^n(\zeta_k^1, i_k + u_k^1) \\ f^n(\zeta_k^2, i_k + u_k^2) \\ \vdots \\ f^n(\zeta_k^N, i_k + u_k^N) \end{bmatrix} \tag{4a}$$

$$y_k = \sum_{m=1}^{N} y_k^n = \sum_{m=1}^{N} g^n(\zeta_k^n, i_k + u_k^n). \tag{4b}$$

III. Problem Formulation and MPC Setup

A. Optimal Control Problem

Since voltage limit is the most important (post-design) factor that impacts the battery operational window, the control objective here is to actively dispatch charge from cell to cell so that all cells stay away from the lowest voltage bound, denoted as $\underline{y}$, below which the cell would fail and lead to the failure of entire pack. In other words, the goal is to find the balancing current u_k^n for $n = 1, \dots, N$ and $k = 0, \dots, K$ so that the cell voltage satisfies

$$\underline{y} \le y_k^n, \quad n = 1, \dots, N \quad \& \quad k = 0, \dots, K, \tag{5}$$

for any driving cycle in the form of current profile, i_k, $k = 0, \dots, K$.

To achieve this goal, we formulate the problem as a model predictive control problem, which uses a short moving horizon to predict the future evolution and optimizes the objective function over this relatively short horizon (compared to the full driving cycle). Denoting $u_k = [u_k^1, u_k^2, \dots, u_k^N]^T$, the optimal control problem (OCP) for MPC to solve at time k is given by

$$\min_{u_k} \quad J(u_k) \tag{6a}$$

$$\text{s.t.} \quad \zeta_{k+j+1}^n = f^n(\zeta_{k+j}^n, i_{k+j} + u_k^n), \quad 0 \le j \le p-1, 1 \le n \le N \tag{6b}$$

$$y_{k+j}^n = g^n(\zeta_{k+j}^n, i_{k+j} + u_k^n), \quad 1 \le j \le p, 1 \le n \le N \tag{6c}$$

$$u_{\min} \le u_k^n \le u_{\max}, \quad 1 \le n \le N \tag{6d}$$

$$\underline{y} \le y_{k+j}^n, \quad 1 \le j \le p, 1 \le n \le N \tag{6e}$$

$$0 = \sum_{i=1}^{N} u_k^n, \tag{6f}$$

where p is the prediction horizon. Note the last constraint (6f) indicates that the balancing circuit is only responsible to

transport charge from one cell to another, and does not provide additional charge and neither is it capable of consuming additional charge (hence different from dissipative balancing strategy). Note also that in OCP (6) the MPC is to optimize one balancing current u_k^n for each cell, which is then kept unchanged over the entire prediction horizon. This strategy is adopted from [9] as the balancing currents are almost constant over the prediction horizon. Note also that the above OCP (6) requires a short-term prediction of the load profile $i_{k+j}, 1 \leq j \leq p$ over the prediction horizon. In case such preview is not available, the value at time k would be used throughout the whole horizon, i.e., $i_{k+j} = i_k, 1 \leq j \leq p$, which is then considered as disturbance to the control system.

B. Objective Functions

Though the general balancing problem we consider is similar to that of [9], the cost function we consider for the OCP (6) would be much different from [9], resulting in completely different control strategies. In particular, we consider three different objective functions for (6a).

In the first formulation, we use a nominal cell model to integrate over the prediction horizon using the requested total current i_k (or i_{k+j} if preview is available), and track each cell's voltage/SOC to follow the nominal cell. The dynamics of the nominal cell are the same as those of (1) but with nominal parameters. Then we integrate the nominal cell model using the initial condition $\zeta_k^0 = \frac{1}{N}\sum_{n=1}^N \zeta_k^n$ to obtain the nominal sequences $\zeta_{k+1}^0, \zeta_{k+2}^0, \ldots, \zeta_{k+p}^0$ and $y_{k+1}^0, y_{k+2}^0, \ldots, y_{k+p}^0$, and the cost function (6a) is defined as

$$J_{\mathrm{t},\sigma}(u_k) = \sum_{j=1}^{p} \left(\sigma_{k+j} - \sigma_{k+1}^0\right)^T \left(\sigma_{k+j} - \sigma_{k+1}^0\right) + u_k^T R u_k. \tag{7}$$

where σ_{k+j} is defined as $\sigma_{k+j} = [\sigma_{k+j}^1, \sigma_{k+j}^2, \ldots, \sigma_{k+j}^N]^T$ and $\sigma \in \{s, y\}$ can be either cell's SOC or terminal voltage, and R is a positive semi-definitive weighting matrix.

Remark 1: Note that the first term of (7) is to penalize the deviation from nominal trajectory while the second term prevents large balancing current that may result in energy waste through resistant heating. Note also that the first term does not require a scaling matrix as in this work the output being tracked is scalar, i.e., either SOC or voltage. The weighting among these two terms can be achieved through the R matrix alone.

In the second formulation, instead of tracking a nominal trajectory, we formulate the MPC to directly maximize the lowest cell SOC or voltage. In other words, the cost function (6a) is defined as

$$J_{\mathrm{m},\sigma}(u_k) = -\sum_{j=1}^{p} \min_n \sigma_{k+j}^n + u_k^T R u_k, \tag{8}$$

which is to *maximize* the lowest cell SOC/voltage for each time step over the prediction horizon with minimal balancing current. In order to reformulate $J_{\mathrm{m},\sigma}$ to be manageable for embedded environment, the trick introduced in [17] is adopted as follows. With addition of p slack variables, $\epsilon = [\epsilon_1, \epsilon_2, \ldots, \epsilon_p]^T$, the objective function (8) can be rewritten as,

$$J_{\mathrm{m},\sigma}(u_k, \epsilon) = -\sum_{j=1}^{p} \epsilon_j + u_k^T R u_k, \tag{9}$$

with additional constraint

$$\epsilon_j \leq \sigma_{k+j}^n, \qquad 1 \leq j \leq p, \quad 1 \leq n \leq N. \tag{10}$$

In the third (and last) formulation, instead of maximizing the lowest cell SOC or voltage, we set up the MPC to minimize the difference between the highest and lowest cell SOC/voltage. Specifically, the cost function (6a) is defined as

$$\begin{aligned} J_{\Delta,\sigma}(u_k) &= \sum_{j=1}^{p} \left(\max_n \sigma_{k+j}^n - \min_n \sigma_{k+j}^n\right) + u_k^T R u_k \\ &= \sum_{j=1}^{p} \max_n \sigma_{k+j}^n - \sum_{n=1}^{p} \min_n \sigma_{k+j}^n + u_k^T R u_k. \end{aligned} \tag{11}$$

Similarly, with a slight abuse of notation, define $2p$ slack variables, $\epsilon = [\epsilon_1, \epsilon_2, \ldots, \epsilon_p, \epsilon_{p+1}, \ldots, \epsilon_{2p}]^T$, the objective function (11) can be rewritten as,

$$J_{\Delta,\sigma}(u_k, \epsilon) = \sum_{j=1}^{p} \epsilon_{p+j} - \sum_{j=1}^{p} \epsilon_j + u_k^T R u_k, \tag{12}$$

with additional constraint

$$\epsilon_j \leq \sigma_{k+j}^n, \qquad 1 \leq j \leq p, 1 \leq n \leq N \tag{13a}$$

$$\epsilon_{p+j} \geq \sigma_{k+j}^n, \qquad 1 \leq j \leq p, 1 \leq n \leq N. \tag{13b}$$

Remark 2: Please note that in the last two formulations, constraints (10) and (13) are only one sided, e.g., $\epsilon_j \leq \sigma_{k+j}^n$ instead of $\epsilon_j \leq \pm\sigma_{k+j}^n$. This is due to the fact that both s^n and y^n are positive by design and hence the cost functions $J_{\mathrm{m},\sigma}$ and $J_{\Delta,\sigma}$ are not based on conventional infinity norm.

In summary, the MPC for the first formulation, denoted as $J_{\mathrm{t},\sigma}$, solves the following OCP

$$\begin{aligned} \min_{u_k} \quad & (7) \\ \text{s.t.} \quad & (6b), (6c), (6d), (6e), (6f). \end{aligned}$$

The MPC for the second formulation, denoted as $J_{\mathrm{m},\sigma}$, solves the following OCP

$$\begin{aligned} \min_{u_k, \epsilon} \quad & (9) \\ \text{s.t.} \quad & (6b), (6c), (6d), (6e), (6f), (10). \end{aligned}$$

The MPC for the third formulation, denoted as $J_{\Delta,\sigma}$, solves the following OCP

$$\begin{aligned} \min_{u_k, \epsilon} \quad & (12) \\ \text{s.t.} \quad & (6b), (6c), (6d), (6e), (6f), (13). \end{aligned}$$

Remark 3: Please note that we can linearize the prediction model (6b) and (6c) for each time step k, around the current

state feedback ζ_k, requested pack current i_k and balancing current at previous control loop u_{k-1}. Therefore, for all three formulations, the OCP (6) can be reformulated into quadratic programming (QP) problem, regardless of which cost function is adopted. QP can be, in general, solved in real time by embedded devices [18], [19], if the problem size is manageable. It is also worth noted that, $J_{\mathrm{t},\sigma}$ has $(2p+1)N$ optimization variables, $J_{\mathrm{m},\sigma}$ has $(2p+1)N+p$ optimization variables with additional pN constraints, while $J_{\Delta,\sigma}$ has $(2p+1)N+2p$ optimization variables with additional $2pN$ constraints. As can be seen, $J_{\mathrm{m},\sigma}$ and $J_{\Delta,\sigma}$ have larger problem sizes and require higher throughput to solve, while at the same time, provide certain benefits in some conditions, as will be seen in the next section.

C. Infeasible Constraint

Note that the output constraint (6e) can be infeasible when the cell voltage is approaching its lower bound. When this happens, we introduce an additional slack variable ϵ_y, and add to each cost function an additional term $W\epsilon_y^2$ where $W \gg R$, and modify the constraint (6e) into

$$\underline{y} \leq y_{k+j}^n + \epsilon_y, \qquad 1 \leq j \leq p, 1 \leq n \leq N \tag{17}$$

In other words, MPC will solve the OCP with original constraint (6e), and towards the end of driving cycle when the OCP is found to be infeasible, MPC will then modify the cost function and replace (6e) with (17) as discussed here.

IV. Simulation Results

In this section, the effectiveness of the proposed MPC formulations will be demonstrated through simulations. In particular, two scenarios are considered. In the first scenario, a constant requested current i_k is considered, which is selected so that the simulation can be conducted in a reasonable amount of time. In the second scenario, the vehicle follows a realistic driving cycle, i.e., FTP cycle, where the vehicle is controlled by an MPC speed tracking controller that requested a battery power P_k [20]. At each time k, P_k is then converted to requested current by $i_k = \frac{P_k}{y_{k-1}}$, where y is defined in (4). Note that in this case, the preview of i_k is assumed to be unavailable. Due to the recent advancement of connected and automated vehicle, the preview of P_k may be estimated with acceptable noise. However, such availability assumption can be too restrictive for the present study.

For each of these two scenarios, the three MPC formulations with $\sigma = y$ will be considered. In other words, the first MPC (denoted as J_{t}) tracks all cells' voltages. The second MPC (denoted as J_{m}) maximizes the lowest cell voltage. And the third MPC (denoted as J_{Δ}) minimizes the difference between the highest and lowest cell voltages. For all setups, $N = 5$ is used and all cells are initialized to be fully charged. The cell parameters V_{oc}^n, R_o^n, R_p^n, and C_p^n are randomly generated to be within 10% deviation from the nominal values.

Setup	Operation Time [s]	Extension
No balancing	1,528	-
J_{t}	1,599	4.65%
J_{m}	1,640	7.33%
J_{Δ}	1,630	6.68%

TABLE I

Simulation results for constant discharge current.

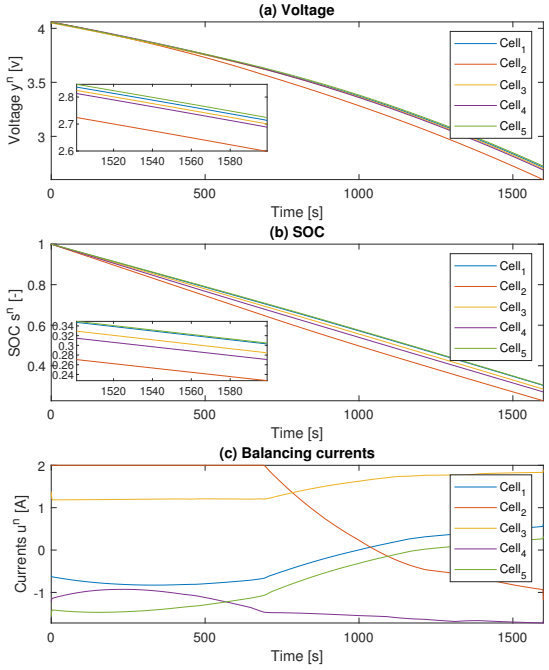

Fig. 2. Results for J_{t} with constant discharge current.

A. Constant Discharge Current

Recall that in this case, constant commanded current i_k is used to represent the steady state operation. Without the active cell balancing, the battery pack can last 1,528 seconds until the lowest cell voltage drops below $\underline{y}$. For MPCs with prediction horizon $p = 5$, J_{t} can extend the operation time to 1,599 seconds (4.65% increase), J_{m} extends to 1,640 seconds (7.33% increase), while J_{Δ} 1,630 extends to seconds (6.68% increase). This is summarized in Table I. Furthermore, Fig. 2 depicts, for the case of J_{t}, each cell's voltage, SOC, and balancing current. It can be seen that the balancing currents are near constant or vary slowly for most of the time, justifying the use of constant balancing current over the prediction horizon. Similar plots for J_{m} and J_{Δ} are omitted due to space limit.

Furthermore, Fig. 3 compares the lowest cell voltage for different controllers, as well as the balancing effort, which represents an index for Ohmic heating loss due to balancing and is calculated as $e_k = u_k^T u_k$. It is clear from Fig. 3(b) that, J_{t} requires larger balancing efforts, especially when the SOC and voltage are still high. This is because J_{t} tracks all cell voltages to the nominal trajectory, and hence will try to balance even when all cell voltages are clear away from the lower bound $\underline{y}$. When the cell voltage gets closer to $\underline{y}$,

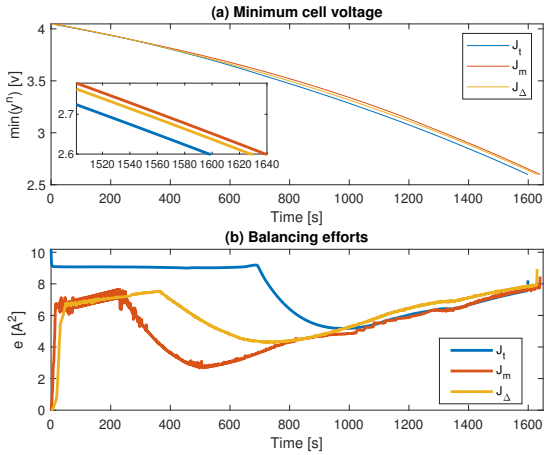

Fig. 3. Comparison of the lowest cell voltage and balancing efforts for different MPC formulations with constant discharge current.

p	J_{t}	J_{m}	J_{Δ}
5	4.3	3.35	8.34
35	28.83	701.36	5309.3

TABLE II

THROUGHPUT COMPARISON FOR DIFFERENT MPC FORMULATIONS. THE UNIT IN THE TABLE IS MILISECOND.

all three MPCs utilize a similar amount of balancing efforts, while J_{Δ} is a little more aggressive.

In addition, we set $p = 35$ and reduce the current to a reasonable level, while at the same time scale the calibration R according to p to balance the two terms in the cost functions. For J_{t} formulation, without balancing, the battery failed at 5 hours, 16 minutes and 45 seconds, while with active cell balancing, it failed at 5 hours, 33 minutes and 52 seconds, providing a 5.13% range extension, which is a bit more than the 4.65% reported in Table I. Note that conducting a similar simulation for J_{m} and J_{Δ} is not possible due to the long simulation time (see Table II).

The throughputs required by each MPC are summarized in Table II, which is measured on a desktop computer with standard CPU using Matlab's standard matrix operations and quadprog as the QP solver.

B. Realistic Driving Cycle

In this section, the vehicle follows a realistic driving cycle, i.e., FTP cycle, where the vehicle is controlled by an MPC speed tracking controller, as presented in [20]. The vehicle speed and power of FTP cycle is then concatenated and scaled up so to provide a realistic assessment of the range extension within a manageable amount of simulation time.

The range extensions for different controllers for $p = 5, 10, 15$, together with their balancing efforts defined as $e = \frac{1}{K}\sum_{k=1}^{K} e_k$, are presented in Tables III. With prediction horizon $p = 5$, all controllers can achieve 4 percent of range extension, with very minimum balancing efforts. For J_{t}, this benefit keeps as p increases, while for J_{m} and J_{Δ} the extension slightly decreases with the increase of p. Such

Setup	p	Distance [m]	Extension	e [A^2]
No balancing	-	89907.12	-	-
J_{t}	5	93507.8	4%	3.91
J_{m}	5	93507.8	4%	0.96
J_{Δ}	5	93507.8	4%	0.4
J_{t}	10	93507.8	4%	15.73
J_{m}	10	93268.78	3.74%	13.33
J_{Δ}	10	93507.8	4%	11.83
J_{t}	15	93494.88	3.99%	15.73
J_{m}	15	93268.78	3.74%	11.33
J_{Δ}	15	93268.78	3.74	11.83

TABLE III

SIMULATION RESULTS FOR FTP CYCLE.

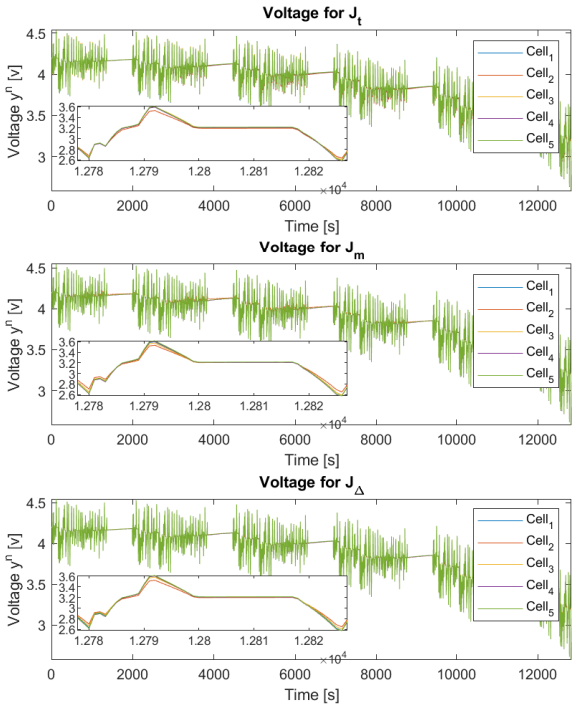

Fig. 4. Cell voltages comparison for different MPC formulations with FTP cycle.

slight decrease may be due to the fact that we are not using preview on load profile in the present simulation, making longer term prediction less effective.

Finally, Fig. 4 plots the cell voltages for the three MPCs, where very similar behaviors are observed. Fig. 5 compares the minimum cell voltages and balancing efforts for a short period of time that is preceding to the pack failure. Though the minimum cell voltages for three MPCs are almost the same in Fig. 5(a), the balancing efforts are very much different. In particular, similar to the steady state scenario, J_{t} requires the maximum amount of balancing efforts.

C. Further Discussion

From Table I it seems that for steady state condition, J_{m} and J_{Δ} can achieve better range extensions with lower

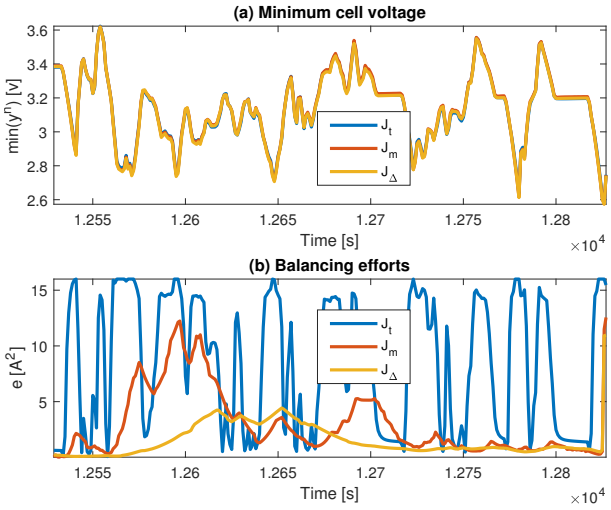

Fig. 5. Comparison of the lowest voltages and balancing efforts for different MPC formulations with FTP cycle.

balancing efforts. However, they require a significant amount of throughput compared to J_t, according to Table II. Therefore, with shorter prediction horizon only, they seem to be better choices for steady state condition. On the other hand, according to Table III, for transient condition, J_t is much more robust against disturbance on future load profile, achieves better range extension with a slightly higher balancing efforts. Therefore, J_t seems to be a better choice for transient condition.

V. Conclusion

In this paper, we studied the active cell balancing problem by using model predictive control for real time range extension. Specifically, three MPC formulations were investigated. In the first formulation, a nominal cell was used to compute a short term reference trajectory and MPC was set to track all cell voltages to follow this reference trajectory. In the second and third formulations, MPC was set to maximize the lowest voltage cell and to minimize the difference between the highest and lowest cell voltage, respectively. To demonstrate the effectiveness of these controllers, both steady state and transient conditions were simulated. In general, a 7% range extension can be achieved for steady state condition, while for transient condition, this is reduced to 4%. It was also found that different driving scenarios may favor different MPC formulations, and a hybrid approach might be needed. Comparing to the existing approaches in literature, our approach can achieve similar range extension without restrictively requiring the final battery state-of-charge to be known in advance. For future work, we would focus on (1) designing an observer to estimate the cells' voltage and SOC, as full state feedback was assumed in the current work, (2) including more driving scenarios to provide a more realistic assessment on the range extensions, and (3) investigating the application of event-triggered MPC [21] to reduce throughput.

References

[1] F. Hoekstra, H. J. Bergveld, and M. Donkers, "Range maximisation of electric vehicles through active cell balancing using reachability analysis," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 1567–1572.

[2] M. Einhorn, W. Roessler, and J. Fleig, "Improved performance of serially connected li-ion batteries with active cell balancing in electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 6, pp. 2448–2457, 2011.

[3] K. Smith, E. Wood, S. Santhanagopalan, G. Kim, and A. Pesaran, "Advanced models and controls for prediction and extension of battery lifetime (presentation)," 2 2014. [Online]. Available: https://www.osti.gov/biblio/1114881

[4] Y. Shang, N. Cui, and C. Zhang, "An optimized any-cell-to-any-cell equalizer based on coupled half-bridge converters for series-connected battery strings," *IEEE Transactions on Power Electronics*, vol. 34, no. 9, pp. 8831–8841, 2018.

[5] C. Pinto, J. V. Barreras, E. Schaltz, and R. E. Araujo, "Evaluation of advanced control for li-ion battery balancing systems using convex optimization," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 4, pp. 1703–1717, 2016.

[6] M. Preindl, "A battery balancing auxiliary power module with predictive control for electrified transportation," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 8, pp. 6552–6559, 2017.

[7] J. Xu, B. Cao, S. Li, B. Wang, and B. Ning, "A hybrid criterion based balancing strategy for battery energy storage systems," *Energy Procedia*, vol. 103, pp. 225–230, 2016.

[8] C. Wang, G. Yin, F. Lin, M. P. Polis, C. Zhang, J. Jiang *et al.*, "Balanced control strategies for interconnected heterogeneous battery systems," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 1, pp. 189–199, 2015.

[9] F. S. Hoekstra, L. W. Ribelles, H. J. Bergveld, and M. Donkers, "Real-time range maximisation of electric vehicles through active cell balancing using model-predictive control," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2219–2224.

[10] J. Liu, Y. Chen, and H. K. Fathy, "Nonlinear model-predictive optimal control of an active cell-to-cell lithium-ion battery pack balancing circuit," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 483–14 488, 2017.

[11] L. McCurlie, M. Preindl, and A. Emadi, "Fast model predictive control for redistributive lithium-ion battery balancing," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 2, pp. 1350–1357, 2016.

[12] Z. Pei, X. Zhao, H. Yuan, Z. Peng, and L. Wu, "An equivalent circuit model for lithium battery of electric vehicle considering self-healing characteristic," *Journal of Control Science and Engineering*, vol. 2018, 2018.

[13] S. S. Madani, E. Schaltz, and S. Knudsen Kær, "An electrical equivalent circuit model of a lithium titanate oxide battery," *Batteries*, vol. 5, no. 1, p. 31, 2019.

[14] B. Y. Liaw, R. G. Jungst, A. Urbina, and T. L. Paez, "Modeling of battery life i. the equivalent circuit model (ecm) approach," Tech. Rep.

[15] H. He, R. Xiong, X. Zhang, F. Sun, and J. Fan, "State-of-charge estimation of the lithium-ion battery using an adaptive extended kalman filter based on an improved thevenin model," *IEEE Transactions on vehicular technology*, vol. 60, no. 4, pp. 1461–1469, 2011.

[16] J. Wehbe and N. Karami, "Battery equivalent circuits and brief summary of components value determination of lithium ion: A review," in *2015 IEEE TAEECE*, 2015, pp. 45–49.

[17] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear model predictive control*. Springer, 2009, pp. 345–369.

[18] A. Bemporad, D. Bernardini, R. Long, and J. Verdejo, "Model predictive control of turbocharged gasoline engines for mass production," SAE Technical Paper, Tech. Rep., 2018.

[19] G. Cimini and A. Bemporad, "Exact complexity certification of active-set methods for quadratic programming," *IEEE Transactions on Automatic Control*, vol. 62, no. 12, pp. 6094–6109, 2017.

[20] J. Chen, M. Liang, and X. Ma, "Probabilistic analysis of electric vehicle energy consumption using mpc speed control and nonlinear battery model," in *2021 IEEE Green Technologies Conference*, Denver, CO, April 7–9, 2021.

[21] J. Chen and Z. Yi, "Comparison of event-triggered model predictive control for autonomous vehicle path tracking," in *2021 IEEE Conference on Control Technology and Applications (CCTA)*, San Diego, CA, August 8–11, 2021.