Many-to-Many Active Cell Balancing Strategy Design

Matthias Kauer[1], Swaminathan Narayanaswamy[1], Sebastian Steinhorst[1],
Martin Lukasiewycz[1], Samarjit Chakraborty[2]

[1] TUM CREATE, Singapore, Email: matthias.kauer@tum-create.edu.sg   [2] TU Munich, Germany, Email: samarjit@tum.de

Abstract—In the context of active cell balancing of electric vehicle battery cells, we deal with circuit architectures for inductor-based charge transfer and the corresponding high-level modeling and strategy development. In this work, we introduce a circuit architecture to transfer charge between arbitrarily many source and destination cells (many-to-many) for the first time and analyze the advantages over one-to-one transfer.

Balancing simulation with numerical solvers remains challenging because of non-differentiable PWM signals, while the search space for high-level strategy design – crucial for time and energy efficiency – becomes even larger. Consequently, we develop a closed-form charge transfer model that extends state-of-the-art approaches and is three orders of magnitude faster than step-size controlled simulation. With an initial algorithm design based on experimentally derived rules, we demonstrate that many-to-many transfer dominates neighbor-only approaches in speed and efficiency even though it requires only one additional switch per circuit module.

I. Introduction

Recently, there has been a growing interest in efficient Electrical Energy Storages (EESs). Electric Vehicles (EVs) and stationary smart grid applications create momentum for high capacity EESs beyond portable devices. Given the high power and energy density requirements, Lithium-Ion (Li-Ion) battery cells are dominating other options under most circumstances. The main drawback is their sensitivity to operating parameters. Irreversible damage can occur from out-of-specification operation. Excessive discharge can permanently decrease the capacity of a cell, over-heating can lead to thermal runaway and thus fire or explosion of the battery pack in the worst case. It has therefore become common to deploy sophisticated Battery Management Systems (BMSs) that monitor and control the cell to ensure it remains within a safe range.

In high voltage applications, a crucial function of the BMS is cell balancing. Individual Li-Ion cells deliver a voltage of only about 4V and limited capacity. Many are thus typically connected in series to achieve the desired capacity and high voltage (around 400V in EVs). Within such series topologies, the overall State-of-Charge (SoC) is determined by the cell with the lowest charge since discharging of the pack cannot continue once the first cell reaches its lower threshold. State-of-the-art approaches incorporate *passive* cell balancing, where the charge of all cells is reduced to the one of the cell with the lowest charge by dissipating excess charge via a switched resistor. Albeit widely adopted for its simplicity, this approach is inefficient from an energy perspective.

Energy efficiency is the main motivation for modern EES implementations, however. *Active* cell balancing architectures are hence emerging which transfer charge between cells instead of dissipating it. This significantly increases both energy efficiency as well as effective capacity compared to passive cell balancing approaches.

Whereas the architectural perspective on active cell balancing, i.e., the design of charge transfer circuits has been extensively studied, quantitative models and large-scale strategies have barely been investigated. Notwithstanding, initial research has shown that the high-level balancing strategy ultimately determines the speed and efficiency of its underlying charge transfer architecture.
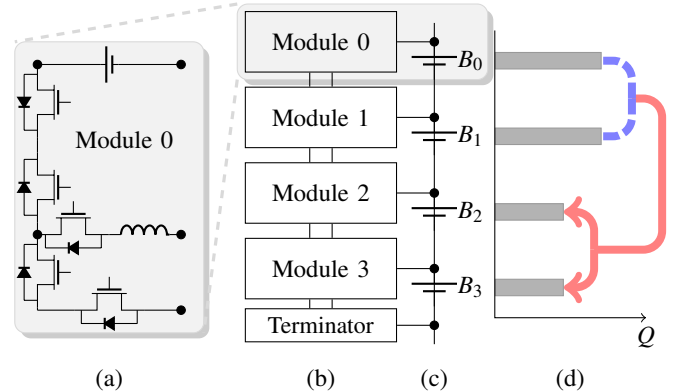
Fig. 1: *After composing individual modules (a) to a balancing architecture (b) and attaching it to the main battery string (c), charge can be transferred from a group of cells to an adjacent group of cells. Here, cells B_0 and B_1 charge their inductor together (━ ━) from where B_2 and B_3 are charged simultaneously (━━). This enables many-to-many transfers between adjacent groups.*

Contributions. This paper presents (i) a new charge transfer circuit that enables transmissions from multiple source cells to multiple adjacent destination cells (see Fig. 1). This significant increase in flexibility – and efficiency as we will see – requires only 5 Metal-Oxide-Semiconductor Field-Effect Transistors (MOSFETs) per module, one more than a neighbor-only transfer circuit. The neighbor-only circuit presented in [1] requires even less, but it routes current over the MOSFET diode which significantly decreases the energy efficiency. On the other hand, the component count of the proposed circuit is decidedly smaller than that of the architecture for non-adjacent transfers presented in [2]. Unlike the circuit there, our new architecture also does not add switches to the path of the main application current and consequently does not influence the standard performance. All the aforementioned circuits are fully modular.

Furthermore, we develop (ii) a rapid simulation framework by extending the charge transfer model from [2]. To the best of our knowledge, many-to-many transfers, as offered by the architecture from Fig. 1, were not covered by state-of-the-art high level models with closed-form solution beforehand. Such formulations are crucial because only the speedup they achieve over straightforward circuit simulation makes the large-scale simulations that we are interested in feasible.

Finally, (iii) new high-level strategies capture the options the proposed circuit offers. They demonstrate the circuit's higher balancing speed and energy efficiency compared to neighbor-only architectures. In fact, the circuit performs at least on par with more complex designs that allow for non-adjacent transfers and vastly outperforms them in certain scenarios.

Organization of the paper. The remainder of the paper is organized as follows. Section II introduces related work concerning the design of charge transfer architectures, respective high-level strategies and their evaluation. Next, Section III introduces the proposed circuit and the Pulse Width Modulation (PWM) signals that drive it. In Section IV, we explain how the circuit can be rapidly simulated before presenting balancing strategy design in Section V. Finally, Section VI shows an experimental evaluation of

both architecture and new strategies before Section VII concludes.

II. RELATED WORK

In order to maintain an equal SoC across all cells of a battery pack, BMSs have to perform cell balancing. Many approaches, which can be classified into *passive* and *active* cell balancing, are reviewed in [3], [4]. Passive cell balancing is the most commonly applied approach as its control and hardware requirements are low. However, it is highly inefficient from an energy perspective since excess charge is just dissipated over resistors and therefore lost.

Active cell balancing, by contrast, transfers charge between cells to reach an equal SoC of all cells conserving most of the energy transferred during balancing. Capacitor-based architectures such as [5] offer simple control design and low installation space but charge transfer via capacitors without involving additional inductive DC/DC converters is not energy efficient. Architectures involving transformers, as presented in [6], can offer charge transfers between non-neighboring cells at a high speed and energy efficiency. However, they suffer from high installation space requirements and hardware cost. Inductor-based architectures are more energy efficient than capacitor-based approaches and require less installation space than those using transformers. Routing charge across one single inductor between cells in a battery is proposed in [7]. This non-modular approach cannot operate concurrently and requires a switching network. By contrast, approaches involving inductors per module can transfer charge concurrently between neighboring cells [1] or non-neighboring cells [2], offering high balancing speed and energy efficiency in a modular architecture.

Finally, [8] shows an initial comparison of balancing strategies for a DC-DC converter architecture. However, the authors limit their case study to 10 cells, assume commutativity, and forgo modeling distance effects.

III. NEW MANY-TO-MANY CHARGE TRANSFER CIRCUIT

In this section, we introduce the proposed balancing architecture which was verified with the approach in [9]. We begin by explaining the underlying circuit and subsequently detail the PWM signals that drive the charge transfer.

Our proposed architecture consists of homogeneous modular charge transfer blocks. Fig. 2 is a more detailed view of Fig. 1 and shows 4 modules attached to the battery cells B_0 to B_3. Each module is assembled from 5 MOSFETs and one inductor surrounding a battery cell. They are terminated on the right by an additional column of 3 MOSFETs[1]. The main path through the

---

[1]It is possible to avoid this termination block and retain full modularity by slightly modifying the switching rules.
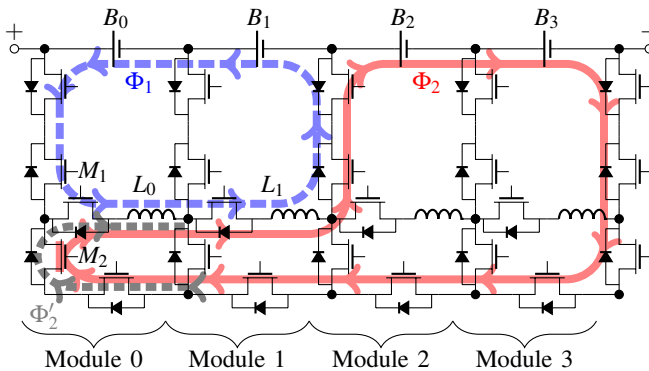
Fig. 2: *The proposed balancing circuit can transfer charge between adjacent groups of arbitrary size. Here, B_0 and B_1 simultaneously charge their inductors (Φ_1) who subsequently discharge into B_2 and B_3 (Φ_2). Φ_2' is an interim safety configuration that briefly routes Φ_2 over a diode to prevent short circuits.*
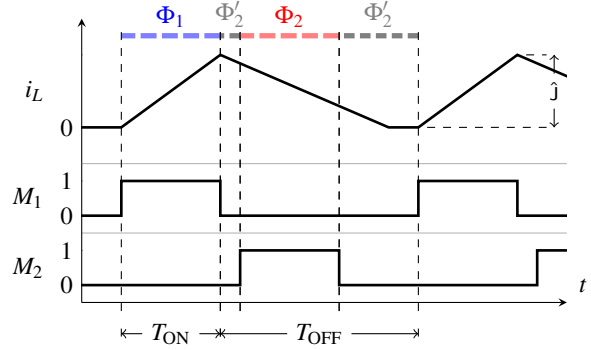
Fig. 3: *Control scheme for the transfer configurations in Fig. 2 and resulting inductor current. Configuration Φ_1 charges the inductors until they reach peak current $\hat{\jmath}$ after a time of T_{ON}. Φ_2 then discharges the inductors until they are entirely empty. T_{OFF} designates the required time for this plus a small addition for safety configuration Φ_2'.*

cells which carries the load current of the battery pack remains transistor-free. This ensures that the operating performance of the EES is not influenced by the balancing hardware except during actual balancing.

Transfers are conducted by operating two transistors in non-overlapping PWM mode. It has been ensured that these are always controlled from within the same module to facilitate local PWM generation. The other MOSFETs operate in static configuration. Fig. 2 depicts a possible transfer scenario that could run on this excerpt of the overall series topology. Φ_1 and Φ_2 represent a transfer from cells B_0 and B_1 to cells B_2, B_3 (two-to-two, driven by M_1 and M_2 of module 0). Configuration Φ_2' is an interleaved variation of Φ_2 that runs over the diode in transistor M_2 to avoid short circuits. With suitable re-configuration of the other transistors module 0 could also drive other transfers such as B_0 to B_1, B_0 to (B_1, B_2) or (B_0, B_1) to B_2.

Fig. 3 details i_L, the current in the inductors of modules 0 and 1 as well as the PWM signals in the aforementioned MOSFETs. M_1's signal is high during time period T_{ON} and charges the inductor until it reaches the peak current $\hat{\jmath}$. This corresponds to configuration Φ_1. In the ensuing period T_{OFF}, M_1 is first switched off before turning on M_2 to avoid a possible short-circuit; this leads to configuration Φ_2' and subsequently Φ_2. Finally, M_2 is turned off creating again configuration Φ_2'. In addition to avoiding short-circuits, this also ensures that the inductor fully discharges over the diode of the MOSFET. The time spent in configuration Φ_2' is extremely short and it can thus often be ignored in practice. Nevertheless, we perform the upcoming calculations of Section IV-B with the diode drop voltage V_d taken into account. More detailed reasoning about these periods can be found in Remark 1.

IV. FAST SIMULATION MODEL

As discussed in Section III, the inductor-based charge transfer circuits we are concerned with are driven by PWM signals. Depending on the hardware components, their frequency ranges from 1kHz to 100kHz. Standard solvers that analyze such a system using a step size controlled simulation approach struggle with the discontinuities of the PWM signal. They greatly reduce the step size before every switching and only slowly increase it again afterwards. We are interested however in simulations of system-level balancing runs until full equalization of up to 100 cells. Balancing time in such runs can reach several hours which is on a vastly different scale than the short PWM periods and the even smaller steps a conventional solver could take. This leads to extremely long simulation times and makes state-of-the-art methods practical only for the independent analysis of individual PWM phases and small numbers of batteries.
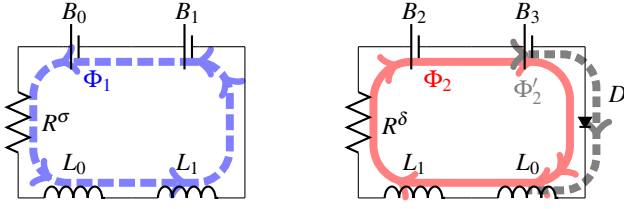
(a) Inductor charging (T_{ON}) (b) Inductor discharging (T_{OFF})

Fig. 4: *By collecting resistances along the current path for configurations Φ_1, Φ_2 and Φ_2' shown in Fig. 2, the charge transfer can be captured by equivalent circuits. As we will show, they are structurally identical for all possible transfers and can thus form the backbone of an analytic charge transfer model.*

In the following, we aim at finding an analytical closed-form solution that transforms the simulation of every PWM into a straightforward calculation instead of the iterative approach a step size controlled algorithm would take. This leads to a simulation speedup of about factor 1000. Using an equivalent circuit model, we derive the transfer losses that occur in the resistances, the inductors and in the batteries whenever charge is moved. Additionally, we consider the switching losses that occur every time a PWM-controlled MOSFET changes its phase. For small inductance L or low peak current $\hat{J}$, the switching losses can become dominating and render the entire process highly inefficient.

A. Switching losses during each PWM step

The switching of a MOSFET dissipates energy due to charging and discharging of input and output capacitances of the transistor in every PWM cycle. This loss is given by $\frac{1}{2}C_{OSS}V_{ds}^2$ where C_{OSS} and V_{ds} are output capacitance and drain-source voltage of the transistor respectively. In addition, the current that is drawn during t_{ON} – summarizing turn-on delay and rise time – and t_{OFF} – consisting of turn-off delay and fall time – cannot be utilized. This entails losses of the form $\frac{1}{2}tI_{ds}V_{ds}$ where I_{ds} is the drain-source current of the transistor. Since I_{ds} vanishes at the beginning of T_{ON} and at the end of T_{OFF}, the switching losses for every single PWM cycle can be characterized by (1) and (2) where σ and δ refer to T_{ON} (source) and T_{OFF} (destination) respectively.

$$E_{sw}^\sigma = \frac{1}{2}t_{\text{OFF}}\hat{J}V_\sigma + \frac{1}{2}C_{OSS}V_\sigma^2 \tag{1}$$

$$E_{sw}^\delta = \frac{1}{2}t_{\text{ON}}\hat{J}V_\delta + \frac{1}{2}C_{OSS}V_\delta^2 \tag{2}$$

Please refer to Chapter 4.3 "Switching Losses" in [10] for further information.

B. Equivalent Circuit Model for Transfer Losses

When the charge transfer involves multiple cells and possibly multiple inductors as in the equivalent circuits shown in Fig. 4, we cannot directly utilize existing charge transfer equations such as the ones proposed in [2]. In the following, we therefore derive a new closed-form solution.

Resistance derivation. On the equivalent circuit level, only the overall resistance resulting from the respective switching configuration needs to be considered. We denote by R^σ and R^δ the resistance during charging phase T_{ON} and discharging phase T_{OFF} respectively. They are composed of inductor resistance R_I, cell resistance R_C and the drain-source on-state resistances of the MOSFETs R_M. In case of the proposed circuit in Fig. 2, we can calculate R^σ and R^δ from Eq. (3).

$$R^\sigma = n^\sigma R_I + (4 + n^\sigma)R_M + n^\sigma R_C$$
$$R^\delta = n^\sigma R_I + (6 + 2n^\sigma + n^\delta)R_M + n^\delta R_C \tag{3}$$

Ordinary Differential Equation (ODE) basis. Getting to the dynamics of the equivalent circuit, we begin by noting that, according to Kirchhoff's Voltage Law, during the T_{OFF} phase (Fig. 4(b)), it has to hold:

$$0 = V_{L,0} + V_{L,1} + V_{R^\delta} + V_{B,2} + V_{B,3} + V_d \tag{4}$$
$$= (L_0 + L_1)\frac{\mathrm{d}}{\mathrm{d}t}i + R^\delta i + \left(\frac{1}{C_2} + \frac{1}{C_3}\right)\int i(\tau)\,\mathrm{d}\tau + V_1 + V_2 + V_d$$

This assumes that the batteries are modeled by a combination of a capacitance C and a resistance R_C. This is justified because, during balancing, the SoC does not overly vary. More about this model can be found in [11]. The T_{ON} phase (Fig. 4(a)) can be treated analogously and we forgo explicitly discussing it here. Note that the constant diode drop voltage V_d (often $= 0$ during T_{OFF} as well, see Remark 1) does not need to be considered at all during T_{ON}.

After introducing abstract inductance $\hat{L}$ and capacitance $\hat{C}$ as

$$\hat{L} = \sum_i L_i \qquad \hat{C}^{-1} = \sum_i C_i^{-1} \tag{5}$$

Eq. (4) can be transformed to Eq. (6) by differentiating it with respect to t. Initial conditions can be derived naturally and have been listed in Table I.

$$\hat{L}\frac{\mathrm{d}^2}{\mathrm{d}t^2}i + R\frac{\mathrm{d}}{\mathrm{d}t}i + \frac{1}{\hat{C}}i = 0$$
$$i(0) = i_0 \qquad \frac{\mathrm{d}}{\mathrm{d}t}i(0) = \mathrm{d}i_0 \tag{6}$$

Along with $\hat{L}$ and $\hat{C}$, we also introduce $\hat{V}$, the abstract voltage and $\hat{Q}$, the abstract equivalent charge as

$$\hat{V} = V_1 + V_2 \qquad \frac{\hat{Q}}{\hat{C}} = V_1 + V_2 = \frac{Q_1}{C_1} + \frac{Q_2}{C_2} \tag{7}$$

In the following, we will (I) show that solving Eq. (6) and using it to calculate $\hat{Q}$ is sufficient and then (II) outline the solution process for Eq. (6).

Charge evolution equivalence. With respect to (I), we observe that if a current flows through the capacitors, it charges all of them equally by ΔQ. It must thus hold

$$\frac{\hat{Q} + \Delta\hat{Q}}{\hat{C}} = \frac{Q_1 + \Delta Q}{C_1} + \frac{Q_2 + \Delta Q}{C_2}. \tag{8}$$

With (7), the definition of $\hat{Q}$, we can rewrite this to

$$\frac{\hat{Q} + \Delta\hat{Q}}{\hat{C}} = \frac{\hat{Q}}{\hat{C}} + \frac{\Delta Q C_2 + \Delta Q C_1}{C_1 C_2}. \tag{9}$$

Multiplying both sides by $\hat{C} = \frac{C_1 C_2}{C_1 + C_2}$, we arrive at

$$\hat{Q} + \Delta\hat{Q} = \hat{Q} + \frac{\Delta Q C_2 + \Delta Q C_1}{C_1 + C_2} = \hat{Q} + \Delta Q \tag{10}$$

and therefore

$$\Delta\hat{Q} = \Delta Q \tag{11}$$

By induction, this relation also holds for $n > 2$ cell capacitances in series. This proves that we can deduce the individual charge evolution from the observation of the equivalent abstract charge $\hat{Q}$.

Charge transfer evolution. Moving on to (II), the solution of ODE (6), we observe that we are still dealing with a *second-order system* since Eq. (4) only has current i, its derivative and anti-derivative as state variables. It can thus be treated using standard methods where the three cases of under-, over- and critical damping

	i_0	$\mathrm{d}i_0$
T_{ON}	0	$V^\sigma / \hat{L}$
T_{OFF}	$\hat{J}$	$-(V^\delta + V_d + R^\delta\hat{J})/\hat{L}$

TABLE I: *Initial conditions for equivalent circuit ODE (6).*

need to be distinguished and treated separately.

On the way to the closed-form charge transfer equations we desire, we first solve (6) and obtain the general solution

$$i(t) = e^{-ct}\left[Ae^{at} - Be^{-at}\right] \tag{12}$$

where

$$A = \frac{\mathrm{d}i_0 - i_0(-\xi\omega_n - \omega_n\sqrt{\xi^2-1})}{2\omega_n\sqrt{\xi^2-1}}, \quad a = -\omega_n\sqrt{\xi^2-1},$$

$$B = \frac{\mathrm{d}i_0 - i_0(-\xi\omega_n + \omega_n\sqrt{\xi^2-1})}{2\omega_n\sqrt{\xi^2-1}}, \quad c = \xi\omega_n,$$

$$\omega_n = \frac{1}{\sqrt{\hat{L}\hat{C}}}, \qquad\qquad \xi = \frac{1}{2}\frac{R}{\hat{L}}\sqrt{\hat{L}\hat{C}}.$$

After integrating (12) depending on the different cases for ξ, we obtain the charge that is transferred during a period T

$$q^{\mathrm{under}}(T) = \frac{-A}{a^2+c^2}\left[c(e^{-Tc}\cos(Ta)-1) - ae^{-Tc}\sin(Ta)\right]$$
$$- \frac{B}{a^2+c^2}\left[a(e^{-Tc}\cos(Ta)-1) + ce^{-Tc}\sin(Ta)\right],$$

$$q^{\mathrm{crit}}(T) = \frac{B}{c^2}\left(1-(Tc+1)e^{-Tc}\right) - \frac{A}{c}\left(e^{-Tc}-1\right),$$

$$q^{\mathrm{over}}(T) = \frac{B}{a+c}\left[e^{-Ta-Tc}-1\right] + \frac{A}{a-c}\left[e^{Ta-Tc}-1\right] \tag{13}$$

for $\xi < 1$, $\xi = 1$ and $\xi > 1$ respectively. This corresponds to under-damped, critically damped and over-damped system configurations. **PWM signal calculation.** T_{ON} and T_{OFF}, the parameters for the timing of the underlying PWM, are difficult to calculate in case of the nonlinear model. We are not aware of a closed-form solution for calculating T_{ON} given $\hat{\jmath}$. It can be calculated iteratively, improving upon the linear estimate $T_{\mathrm{ON}} = \hat{\jmath}\hat{L}/V^\sigma$ where V^σ is the voltage of the sending cells. Alternatively, it can be chosen arbitrarily as long as $\hat{\jmath}$ does not become too large. $\hat{\jmath}$ can then be calculated from (12) for the determination of T_{OFF}. T_{OFF} can be calculated by requiring that the current vanishes. This leads to

$$T_{\mathrm{OFF}}{}^{\mathrm{under}} = \frac{1}{a}\arctan\left(-\frac{A}{B}\right) \qquad T_{\mathrm{OFF}}{}^{\mathrm{crit}} = \frac{-A}{B},$$

$$T_{\mathrm{OFF}}{}^{\mathrm{over}} = \frac{1}{2a}\log\left(\frac{B}{A}\right) \tag{14}$$

for $\xi < 1$, $\xi = 1$ and $\xi > 1$ respectively. Once ξ is determined, simulation can now be performed using (13). In this work, we derive the PWM timing by calculating T_{ON} iteratively and then obtain T_{OFF} from (14).

Many-to-many simulation summary. This concludes the individual charge evolution equations. Overall, we can thus simulate the system as follows. We adjust the inductance and capacitance via Eq. (5). We calculate the abstract charge $\hat{Q}$ as defined in Eq. (7). Eq. (11) guarantees that it is sufficient to track the evolution of $\hat{Q}$. We simulate the charge transfer using Eqs. (13), (14). At the end, we update the individual charges Q_i by adding $\Delta\hat{Q} = \Delta Q$ as observed in $\hat{Q}$.

Remark 1: *Since T_{ON} may not be precisely calculated in practice, there is often a small safety period included in T_{OFF} during which the inductor is discharged over the internal diode of the MOSFET (configuration Φ_2' in Fig. 3). We have included V_d in the calculation here because it does not significantly complicate the derivation steps. This allows for a direct inclusion of Φ_2' by separating the integration of Eq. (12) in three phases. However, since configuration Φ_2' only occurs briefly (around 2μs with proper design) and we employ switching periods ($T_{\mathrm{ON}}+T_{\mathrm{OFF}}$) at least 100 times larger, it does not materially influence the system evolution. Hence, we do not perform this separation here.*

Remark 2: *Real battery cells cannot store 100% of the charge they receive. This can be modeled by introducing charging efficiency*
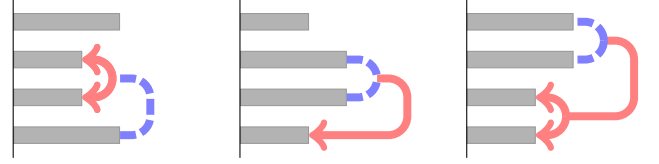
(a) One-to-Two   (b) Two-to-One   (c) Two-to-Two

Fig. 5: *Four scenarios have been selected for fair comparison of local charge transfer options. Scenarios (a), (b), (c) and (d) an unvisualized One to One scenario all begin with the same charge distribution and only end after complete equalization.*

parameter μ and simulating with

$$\tilde{q}_\delta(T_{OFF}) := \mu \cdot q_\delta(T_{OFF})$$

instead of $q_\delta(T_{OFF})$ itself.

Remark 3: *The accuracy of the linear version mentioned in [2] deteriorates with higher resistance which occurs, e.g., during larger many-to-many transfers.*

V. BALANCING STRATEGY DEVELOPMENT

The proposed architecture (Fig. 2) introduces countless possible charge routing configurations and searching for a time- or energy-optimal high-level balancing strategy thus becomes a difficult problem. Even after peak current $\hat{\jmath}$ and a decision period T_D are fixed, we are still looking at an immense search space. If we were to discretize the charge of the cells in n levels, e.g., for a dynamic programming approach, we would end up with up to n^{100} points. At the same time, the charge transferred during individual PWM cycles is small, so the one-dimensional discretization cannot be coarse.

In this work, we mainly aim to demonstrate the potential of the many-to-many transfer architecture and focus on the development of heuristics that achieve this goal. Toward this, we first determine which local charge transfer constellations are beneficial for the overall efficiency in Section V-A. Subsequently, we apply the most beneficial among them as often as possible in the heuristic we propose in Section V-B.

A. Experiments for heuristic development

In this section, we look for the most beneficial local charge transfer constellations. We evaluated the following scenarios that are also depicted in Fig. 5:

(a) One-to-two. Cell 0 transfers to cells 1 and 2 for T_{alt}, starting from a configuration $[V_{\mathrm{high}} \quad V_{\mathrm{low}} \quad V_{\mathrm{low}} \quad V_{\mathrm{high}}]$. Next, cell 3 takes over and also charges cells 1 and 2 for T_{alt}. This cycle is repeated until all cells are equalized.

(b) Two-to-one. With $[V_{\mathrm{low}} \quad V_{\mathrm{high}} \quad V_{\mathrm{high}} \quad V_{\mathrm{low}}]$ as initial configuration, cells 1 and 2 transfer to cell 0 for a duration of T_{alt}. They subsequently switch and transfer to cell 3 for the same length of time. Again, the cycle is repeated until all cells are equalized.

(c) Two-to-Two. Let the cells initially be configured with $[V_{\mathrm{high}} \quad V_{\mathrm{high}} \quad V_{\mathrm{low}} \quad V_{\mathrm{low}}]$. Cells 0 and 1 then transfer to cells 2 and 3 until the pairs (and thus all cells) are balanced.

(d) One-to-One (not shown in Fig. 5). In voltage configuration $[V_{\mathrm{high}} \quad V_{\mathrm{low}} \quad V_{\mathrm{high}} \quad V_{\mathrm{low}}]$, cell 0 begins balancing with cell 1. Once they are equalized, cell 2 then balances with cell 3.

Note that these scenarios have been designed to ensure a fair comparison between the different strategies. All of them start with the same overall energy and finish in a homogeneous voltage configuration.

For our simulation, we chose $[V_{\mathrm{high}} \quad V_{\mathrm{low}} \quad T_{\mathrm{alt}}] = [3.7V \quad 3.2V \quad 10s]$. Basic parameters of the transfer circuit were

$$\begin{bmatrix} C & V_d & R_C \\ R_M & \mu & \hat{\jmath} \end{bmatrix} = \begin{bmatrix} 1\mathrm{e}3\mathrm{F} & 0 & 0.05\Omega \\ 40\mathrm{m}\Omega & 0.97 & 1.0\mathrm{A} \end{bmatrix} \tag{15}$$
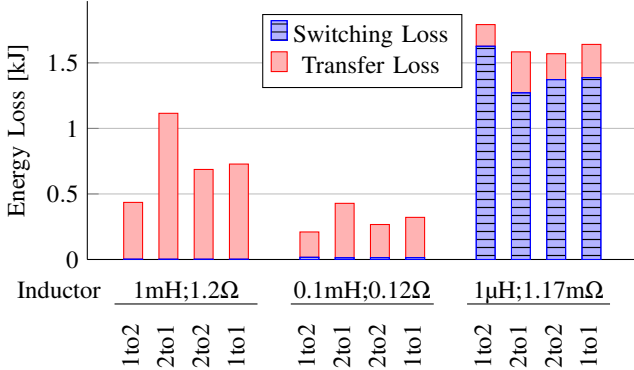
Fig. 6: *Three inductors complete the scenarios from Fig. 5. Sending from one to two appears beneficial for the large and the medium inductor model. The small inductor entails mostly switching losses and profits from combining senders.*

To compute the switching losses, we also needed the following parameters[2]

$$[C_{OSS} \quad t_{ON} \quad t_{OFF}] = [125\text{pF} \quad 44\text{ns} \quad 168\text{ns}] \tag{16}$$

We then evaluated the charge transfer for three representative off-the-shelf inductors with $(L \quad R_I) = (1\text{mH} \quad 1.2\Omega)$, $(0.1\text{mH} \quad 0.12\Omega)$ and $(1\text{μH} \quad 1.17\text{mΩ})$ respectively.

The results of the experiment are shown in Fig. 6. We observe that both for the large and the medium inductor one-to-two transfer is beneficial with over 50% less energy dissipation. For the small inductor, however, it is somewhat favorable to send from multiple cells to single cells. The reason for this behavior becomes more obvious when considering transfer and switching losses separately. The high resistance of the large inductor leads to dominating dissipative losses (over 99%) which are increased by the even higher resistance of multiple inductors. The low inductance of the small inductor on the other hand entails a very high PWM frequency and thus primarily creates switching losses (80–90%). Higher inductance from multiple inductors reduces these.

Note that there is a significant absolute difference between the individual inductors. In fact, Fig. 6 seems to suggest that the medium inductor dominates the other two options. This is not entirely true because we have ignored both installation volume as well as maximum current (and thus balancing time) in this study. Selecting the optimal inductor overall is beyond the scope of this work; an evaluation of off-the-shelf components for balancing purposes was done in [13].

B. Balancing Heuristics

The charge transfer architecture presented in this work opens up new possibilities in system-level strategy design. The transfer pairs to be chosen are no longer just pairs of cells, but pairs of tuples or sequences of cells. This significantly enlarges the design space for suitable strategies. We utilize the following heuristics to demonstrate the potential of the architecture. KUTKUT and SLOW are variants of the algorithm described in [2]. KUTKUT, a neighbor-only method, represents the main competition since it can run on a charge transfer circuit with complexity similar to the one we propose (Fig. 2). SLOW is significantly more efficient than KUTKUT, but it should be considered an *unfair comparison* in the present context as explained in Remark 4. Both find the largest cell and transmit to the smallest cell within a certain range (1 for KUTKUT, #cells for SLOW). KUTKUT transmits concurrently as much as possible while SLOW only performs a single charge transfer at a time. See [2] for details.

---

[2]This data corresponds to a MOSFET which is suitable for this application (see [12]). R_M, the drain-source on-state resistance has been increased from 27 to 40mΩ to account for cable and trace resistances.

Algorithm 1 Transfer tuple selection algorithm MMFIND selects the strongest cell and its strong neighbors as sources. They send in direction of weaker mean. If cells behind first recipient are weak, the destination list is extended. Selection is repeated periodically until cell string Q is equalized.

IN: Charge array Q, maximum concurrent transfers K, expansion parameters $(\kappa \quad \eta)$, desired maximum level of distortion ε
OUT: List of transfer tuples $\mathscr{P}$ or BALANCED
1: **if** $\text{sdev}(Q)/\text{mean}(Q) \leq \varepsilon$:
2:      Return BALANCED
3:    $\mathscr{P} = \{\}, \mathscr{V} = \{1,\dots,N\}$ //Init return, selectable cells
4: **while** $\mathscr{V} \neq \{\} \wedge |\mathscr{P}| < K$:
5:      $s = \arg\max_{j \in \mathscr{V}} Q^{(j)}$
6:      $\mathscr{S} = \left\{ \hat{\sigma} = \{\dots,s,\dots\} \in \mathscr{V} : \frac{Q^{(j)}-\bar{Q}}{Q_{\max}-\bar{Q}} > \kappa \quad \forall j \in \hat{\sigma} \right\}$
7:      $\sigma = \arg\max_{\hat{\sigma} \in \mathscr{S}} |\hat{\sigma}|$ //Find more high charge
8:      $\bar{Q}_< = \text{mean}\left((Q^{(j)})_{j<\min(\sigma)}\right)$
9:      $\bar{Q}_> = \text{mean}\left((Q^{(j)})_{j>\max(\sigma)}\right)$
10:    $\text{dir} = \text{signum}(\bar{Q}_< - \bar{Q}_>)$ //Send to less overall charge
11:    $s_{\text{beg}} = \text{dir} \cdot \max\{\text{dir} \cdot s : s \in \sigma\}$ //head source cell
12:    $\mathscr{D} = \left\{ \hat{\delta} = \{s_{\text{beg}}+2\text{dir},\dots\} \in \mathscr{V} : Q^{(s)} < \eta\bar{Q} \quad \forall s \in \hat{\delta} \right\}$
13:    **if** $s_{\text{beg}} + \text{dir} \notin \mathscr{V}$:
14:      $\mathscr{V} = \mathscr{V} \setminus \{s\}$; Continue
15:    $\delta = \{s_{\text{beg}}+\text{dir}\} \cup \arg\max_{\hat{\delta} \in \mathscr{D}} |\hat{\delta}|$ //Find more weak
16:    $\mathscr{P} = \mathscr{P} \cup \{(\sigma,\delta)\}$
17:    $\mathscr{V} = \mathscr{V} \setminus \{\min(\sigma,\delta)-1,\dots,\max(\sigma,\delta)+1\}$

Remark 4: *Balancing strategy* SLOW *performs transfers between non-adjacent battery cells with an architecture proposed in [2] which is not possible with the architectures from Fig. 2 or [1]. However, to implement the hardware proposed in [2], six switching MOSFETs and two larger, high current MOSFETs are required. Note that the diagrams in [2] omit two MOSFETs that are required to block the conduction of the internal body diode.*

We propose MMSLOW, an adaptation of SLOW for the many-to-many architecture from Fig. 2. Instead of sending from the strongest to the weakest cells directly – impossible on simpler architectures – it transmits to the weakest cell and all cells on the path between them. This mainly leads to one-to-many transfers that are beneficial with larger inductors as we have seen in Fig. 6.

MMFIND is a new heuristic tailored specifically to the opportunities the architecture from Fig. 2 offers. It is listed as Algorithm 1 and begins by trivially initializing the list of transfer tuples $\mathscr{P}$ – the main result and the remaining selectable cells in line 3. It selects as senders the longest sequence of strong cells – quantified as having charge significantly larger than average – containing s, the strongest cell remaining (line 7). κ, the parameter for determining strong should be chosen rather strict to ensure that we rarely create many-to-one transfers (cf. Fig. 6). Subsequently, the direction of less mean charge is found (line 10) to ensure that charge is only transferred in the correct global direction. If the next cell in this direction is still available for transfer (line 13), it is part of the receiver tuple δ (line 15). Additionally, any cell behind it that is sufficiently weak is added as well. η, the parameter that creates the criterion for weak cells, can be somewhat lenient. After all, one-to-many transfers are desirable (cf. Fig. 6). Finally, the resulting tuples σ and δ are added to the return list (line 16) and removed from the selectable cells (line 17). This selection is repeated until the battery pack has reached a satisfactory level of distortion ε.

VI. CASE STUDY

Runtime. On a PWM-level, the calculation time for 200 PWM cycles on a 4 cell architecture varied between 0.25s and 0.33s (0.254s on average) over 100 runs. Compared to a SPICE simulator run requiring 260s on average, our method is therefore around

260s/0.254s = 1023.6 times faster. The relative error remained below 0.3%. As an indication for larger simulations, note that the 80 balancing runs yielding Figs. 7 and 8 ranged from 5min for the random MMSLOW to 35min for the sorted KUTKUT.

Experimental setup. To demonstrate the value of the newly proposed architecture, we will in the following compare the equalization of 100 randomly initialized battery cells using the strategies discussed in Section V-B with respect to time and total energy required. KUTKUT and SLOW are running on their respective architectures with the parameters described before. MMSLOW and MMFIND run on the architecture from Fig. 2. Parameters for MMFIND were $K = 1$, $(\kappa \quad \eta) = (0.85 \quad 1.1)$.

Component-wise, we assumed the basic parameters from (15). The MOSFET has further switching parameters that were described in (16). Finally, we assumed the medium inductor from Section V-A with $(L \quad R_I) = (0.1\text{mH} \quad 0.12\Omega)$.

Random charge distribution. The cells are repeatedly randomized with $Q \sim \mathcal{N}\left(3.6\text{V}, \sqrt{0.05}\right)$. All strategies then fully balance the same configuration and record their performance with respect to time and total energy (switching and transfer losses) requirement. Next, the cells are randomized again and the procedure is repeated. This results in the data shown in Fig. 7.

SLOW performs well in this scenario achieving the fastest balancing time. FAST (from [2]) typically achieves even higher speed at the expense of additional energy losses. Even SLOW dissipates more energy than the new MMSLOW, however, and – most importantly – neither SLOW nor FAST are a fair comparison because they require a vastly larger circuit architecture (see Remark 4). The new MMFIND is quite slow (about 100 % slower than SLOW) and dissipates more energy (about 50%). KUTKUT is on par speed-wise with the faster results, but its transfer losses are significantly higher (around 200%).

Realistic charge distribution. In a real implementation, however, the weak cells wouldn't be randomly distributed. Typically, the location of the cooling inside an EV determines where the strong cells are after a while. We assumed that the cooling was mainly on one side of the string and thus sorted the cells before we began the balancing operation. The other simulation parameters remained unchanged. The results of this scenario are shown in Fig. 8.

MMFIND dominates the other strategies – including the non-competing SLOW – under these starting conditions. This is a very clear indication for the value of the proposed approach. SLOW, in fact, is even dominated by the strategy MMSLOW in this scenario and leads to about 100% additional balancing time and dissipated energy. Finally, KUTKUT severely under-performs under these circumstances. It requires about 5 times more time and loses 5 times more energy than MMFIND.

Interpreting Fig. 7 and Fig. 8, i.e., both scenarios, together, we can conclude that the addition of many-to-many transfer capabilities is highly valuable. In both scenarios, there is a favorable strategy that achieves a balanced string in a highly more efficient fashion than KUTKUT, the strategy with similar hardware
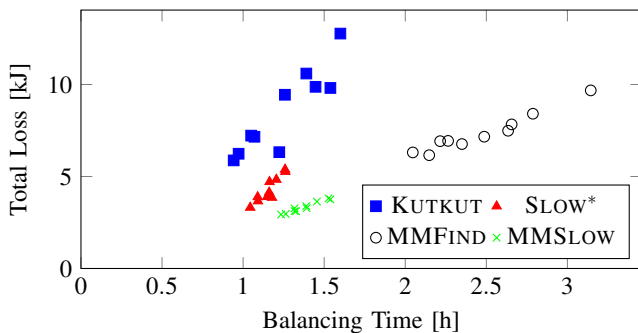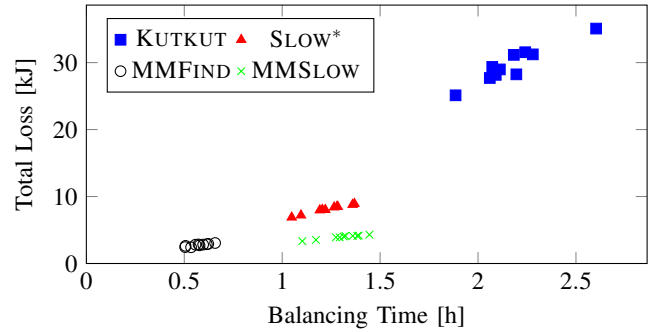
Fig. 8: *Certain operating conditions, such as an unequal cooling, lead to a sloped charge distribution in the battery cell string. We simulated the balancing of 100 randomly initialized and subsequently sorted cells and recorded the total energy consumed during the transfer and the required time. (*: not competing; see Remark 4.)*

requirements. The favorable method also keeps up speed-wise even though, unlike KUTKUT, it doesn't resort to concurrent balancing. The architecture performs on par with the *unfair competitor* (see Remark 4) SLOW in Fig. 7 and dominates everyone in the realistic scenario shown in Fig. 8.

VII. CONCLUDING REMARKS

This paper introduced a new many-to-many charge transfer architecture, presented a suitable closed-form solution for simulation and examined high-level strategies for active cell balancing. The circuit allows transmissions between groups of adjacent cells which renders it highly efficient but also obsoletes current state-of-the-art analytical methods. Hence, we have proposed a new simulation approach in this work that remains over 1000 times faster than straightforward step-size controlled solvers. With appropriate strategies, such as the ones introduced in Section V, the new hardware can dominate similarly complex architectures and performs at least on par with another significantly larger circuit. The individual strategy selection depending on the charge distribution scenario as well as a potential architecture that implements many-to-many capabilities along with non-adjacent transfers shall be investigated in the future.

REFERENCES

[1] N. H. Kutkut, "A Modular Nondissipative Current Diverter for EV Battery Charge Equalization," in *Proc. of APEC*, 1998.
[2] M. Kauer, S. Naranayaswami, S. Steinhorst, M. Lukasiewycz, S. Chakraborty, and L. Hedrich, "Modular System-Level Architecture for Concurrent Cell Balancing," in *Proc. of DAC*, 2013.
[3] S. Moore and P. Schneider, "A Review of Cell Equalization Methods for Lithium Ion and Lithium Polymer Battery Systems," *SAE Technical Paper Series*, 2001.
[4] J. Cao, N. Schofield, and A. Emadi, "Battery Balancing Methods: A Comprehensive Review," in *Proc. of VPPC*, 2008.
[5] A. Baughman and M. Ferdowsi, "Double-Tiered Switched-Capacitor Battery Charge Equalization Technique," *IEEE Trans. on Industrial Electronics*, vol. 55, no. 6, 2008.
[6] D. V. Cadar, D. M. Petreus, and T. M. Patarau, "An Energy Converter Method for Battery Cell Balancing," in *Proc. of ISSE*, 2010.
[7] S.-H. Park, T.-S. Kim, J.-S. Park, G.-W. Moon, and M.-J. Yoon, "A New Buck-boost Type Battery Equalizer," in *Proc. of APEC*, 2009.
[8] F. Baronti, R. Roncella, and R. Saletti, "Performance Comparison of Active Balancing Techniques for Lithium-Ion Batteries," *Journal of Power Sources*, vol. 267, pp. 603 – 609, 2014.
[9] M. Lukasiewycz, S. Steinhorst, and S. Narayanaswamy, "Verification of balancing architectures for modular batteries," in *Proc. of CODES+ISSS*, 2014.
[10] R. W. Erickson and D. Maksimovic, *Fundamentals of Power Electronics*. Springer, 2001.
[11] V. Johnson, "Battery Performance Models in ADVISOR," *Journal of Power Sources*, vol. 110, no. 2, pp. 321 – 329, 2002.
[12] N. Semiconductors, "PMN25UN 20 V, 6 A N-channel Trench MOSFET Product data sheet," 28 July 2011. [Online]. Available: www.nxp.com/documents/data_sheet/PMN25UN.pdf
[13] S. Narayanaswamy, S. Steinhorst, M. Lukasiewycz, M. Kauer, and S. Chakraborty, "Optimal Dimensioning of Active Cell Balancing Architectures," in *Proc. of DATE*, 2014.

Fig. 7: *After balancing 100 cells, the required time and the dissipated energy are recorded for various random initializations. (*: SLOW is not competing; see Remark 4.)*