CrossMark

Improvement of accuracy of multi-scale models of Li-ion batteries by applying operator splitting techniques

Z. Farkas [a,c,*], I. Faragó [a,b], Á. Kriston [c], A. Pfrang [c]

[a] MTA-ELTE NumNet Research Group, Pázmány Péter s. 1/C., H-1117 Budapest, Hungary

[b] Eötvös Loránd University, Department of Applied Analysis and Computational Mathematics, Pázmány Péter s. 1/C., H-1117 Budapest, Hungary

[c] European Commission, Joint Research Centre, Institute for Energy and Transport, Westerduinweg 3, NL-1755 ZG Petten, The Netherlands

ARTICLE INFO

ABSTRACT

In this work operator splitting techniques have been applied successfully to improve the accuracy of multi-scale Lithium-ion (Li-ion) battery models. A slightly simplified Li-ion battery model is derived, which can be solved on one time scale and multiple time scales. Different operator splitting schemes combined with different approximations are compared with the non-splitted reference solution in terms of stability, accuracy and processor cost. It is shown, that the reverse Strang–Marchuk splitting combined with the implicit scheme to solve the diffusion operator and Newton method to approximate the non-linear source term can improve the accuracy of the commonly applied vertical (sequential) multi-scale models by almost 3 times without considerably increasing the processor cost.

1. Introduction

Li-ion batteries (LIB) are electric power sources which have been commercialized to energize portable devices in the 90s by Sony [1]. In recent years LIBs have become the key enabling technology to store sufficient amount of electricity for future electric vehicles (EV) with a desirable driving range. Despite the improved energy storage and power capability of EVs, the battery technology needs to be improved further. Product optimization needs the application of complex mathematical models which can couple diffusion, heat transfer, and migration at molecular level and in the same time explain current, potential, heat and concentration distribution at macroscopic-level.

The mathematical description of LIBs is based on a non-linear parabolic system of partial differential equations with non-linear source terms and couples physical and chemical phenomena from nano to cm scale. This broad length and time scales makes the LIB models difficult to solve. Since no general analytical solution has been derived, several numerical approaches are developed. Franco et al. concluded [2] that there is no single numerical mathematical model which can describe all the coupled phenomena, and suggested [2,3] the application of multi-scale modeling (also known as integrated multi-scale modeling). Multi-scale modeling is based upon the recognition that the types of physics occurring at small time and length scales are distinct from those occurring at longer time and length scales [4]. An effective and accurate multi-scale

---

* Corresponding author at: MTA-ELTE NumNet Research Group, Pázmány Péter s. 1/C., H-1117 Budapest, Hungary.
 E-mail address: farkas.zeno92@gmail.com (Z. Farkas).

model, which couples continuum equations (diffusion, heat transfer, migration) describing current, potential, and concentration distributions with molecular-level events can be used not only to design and to control complex electrochemical system but also to simulate local materials failure events and their impact on global scale behavior (i.e., safety). Consequently, the performance (accuracy, processor cost etc.) of novel models capable to solve multi-scale problems need to be compared and investigated for possible improvement.

Multi-scale models can be vertical, such that the smaller scale physics models (which may be atomistic or microstructural) are embedded and run "inside" the larger size scale physics models (macrohomogeneous). Alternatively, multi-scale models can be horizontal [5], in such a way that the results of lower size scale simulations provide input parameters, e.g. kinetic constants or thermodynamic quantities, for the higher-scale simulations.

In spite of the broad application of multi-scale models, the numerical characteristics (stability, convergence, accuracy) of splitting and embedding the results of sub-problems at different length and time scale require further detailed studies.

Operator splitting techniques are commonly applied for solving complex systems. The main idea is to split the complex problem into a sequence of sub-problems with simpler structure. Hence, the approach of operator splitting and multi-scale modeling is similar in respect of splitting the complex problem to sub-problems. In case of multi-scale modeling they can split the complex multi-scale problem into simpler single scale sub-problems by solving each of the latter by the most appropriate numerical approximation. Several ways to split a complex system into simpler problems and to solve them on different time and length scales have been developed e.g. sequential splitting, symmetrical splitting or Strang–Marchuk (S–M) splitting [5–8]. These methods differ from each other in time discretization providing different computational benefits and drawbacks.

Faragó at al. [6] used sequential splitting method for air pollution modeling and also for the solution of the Maxwell equations including a source term [7]. In both cases accuracy and adaptability of the operator splitting were examined. Kriston et al. [8] applied sequential and symmetrical splitting methods for the simulation of the transient behavior of fuel cells. The applied partial differential system comprised only one dependent variable and two operators. In the case of LIB model, there are at least 3 dependent variables and 4 different operators, therefore they represent a more complex problem. Moreover none of the works [2,4–8] applied operator splitting for multi-scale modeling, and we are not aware of any work which applied operator splitting for multi-scale simulation of LIBs.

In this paper a novel approach for multi-scale modeling of LIBs is developed. Decoupling of the processes at different scales is realized by operator splitting techniques. The mathematical accuracy and processor cost of the developed multi-scale models are analyzed and compared. The results indicate that the accuracy of both of the horizontal and vertical multi-scale models can be substantially improved by an adequately constructed splitting scheme.

On one hand we recommend this paper to mathematicians who would like to start with the simulation of LIBs. On the other hand we detail the mathematical descriptions, algorithms, numerical methods and derivation of models for engineers and other scholarly readers.

1.1. Description of operator splitting

The mathematical model of a LIB can be described in the form of the following abstract Cauchy problem for $t \in [0, T]$ and $x \in [0, L]$

$$\begin{cases} \dfrac{\partial w(\cdot, t)}{\partial t} = \sum_{i=1}^{n} A_i w(\cdot, t) \\ w(\cdot, 0) = w_0(\cdot), \qquad \dfrac{\partial w(0, t)}{\partial x} = g_1(t), \qquad \dfrac{\partial w(L, t)}{\partial x} = g_2(t) \end{cases} \tag{1}$$

where $w : \mathbb{R} \times \mathbb{R} \to \Lambda$ is the Λ-valued unknown function for every fixed $t \in (0, T]$ and Λ denotes the possible states space, which is usually assumed to be a Banach space. Furthermore $w_0(x) \in \Lambda$ and $g_1(t), g_2(t) \in \Lambda$ define the initial and boundary conditions of the problem and operators $A_i : \Lambda \to \Lambda$ define the different sub-processes.

Operator splitting techniques were developed to find the solution of problem (1), when A_i consists of non-linear operator(s). Usually operators are splitted by the different mathematical structures (e.g. linear and non-linear part of the equation are grouped separately) or by the same partial differential operators (grouping different time and space derivatives together), but the splitting is arbitrary. Then the obtained simpler systems are discretized on potentially different meshes. One of the main advantages of operator splitting techniques is that different numerical schemes and discretizations with different length and time scales can be applied, selecting the most adequate one for a given sub-problem. The main drawback, however may be the loss of convergence and accuracy. Multi-scale models also split A_i in respect of the physical processes and solve them at different time and/or length scales, consequently they apply a sort of operator splitting technique. Vertical multi-scale models are most likely similar to a sequential splitting scheme, which is the most simple operator splitting technique.

In this work the separation of the linear and non-linear processes is applied for the splitting of operators. The effect of operator splitting methods (e.g. sequential, symmetrical, S–M), and numerical schemes (e.g. explicit, implicit) on the accuracy are analyzed in detail.

1.2. The accuracy of operator splitting

One of the objectives of this work is to study the accuracy of multi-scale models and to improve it by using an adequately chosen operator splitting scheme. The accuracy of multi-scale models is analyzed by the technique developed for operator splitting by Csomos [9] and Faragó [10]. In the paper [11] there is the general description of the operator splitting process, and the error caused by the splitting is analyzed in detail. In [12] the different errors of the numerical process are introduced, and their relations are given. In [13] the convergence of the different splittings (sequential, S–M) are investigated for several operators with special properties. The convergence of the general weighted sequential splitting method with special properties is shown in [9] including the convergence estimation of symmetrical splitting. Faragó et al. [14] detailed the theoretical convergence order of various splitting methods. It was found that sequential splitting method is a first order, while symmetrical and S–M splitting are second order methods. Consequently, this work's natural aim is to enhance the stability and accuracy of the usually applied sequential method in multi-scale modeling of LIBs by developing a second order splitting algorithm.

The overall convergence of the numerical solution does not only depend on the splitting accuracy. The numerical schemes and types of discretization are other sources of inaccuracy. The most commonly used numerical scheme is the finite difference method (FDM) that approximates the solution in discretized points (mesh points). For instance the convergence condition of explicit FDM of a parabolic partial differential equations (PDE) problem (1) in 1D can be given by the proportion of spatial (denoted by h) and time (denoted by $\Delta\tau$) discretization step sizes, as follows [15]

$$\frac{\Delta\tau}{h^2} \leq \frac{1}{2}$$

Obviously, the overall error (denoted by Err_{total}) is a rough lower theoretical estimate of the splitting and FDM errors, $Err_{x,sp}$, $Err_{x,FDM}$, respectively, which can be calculated by using the triangle inequality, namely:

$$Err_{x,total} \leq Err_{x,FDM} + Err_{x,sp}. \tag{2}$$

Consequently, if a splitting method and a numerical approximation (FDM) are convergent the splitted numerical method is convergent in the appropriately selected norm. While the analytical solution of the problem is not known, neither the splitting nor the discretization inaccuracy can be calculated. Therefore in this paper an adequate mathematical model is developed, which can be solved numerically by a non-splitted and a splitted way. The reference non-splitted solution $w_{x,non-splitted}$ is obtained by Matlab's "pdepe" solver by using a very small time step, while the splitted solution $w_{x,splitted}$ is obtained by using an operator splitting technique and one or more numerical schemes (e.g. explicit, IMEX). The total error therefore can be calculated by the following expression

$$Err_x(\tau) := \|w_{x,non-splitted}(\tau) - w_{x,splitted}(\tau)\|_\Lambda \tag{3}$$

where $\|\cdot\|_\Lambda$ denotes an arbitrary norm. In this paper the classical l_2 norm is used to describe the overall accuracy in the whole time domain, while the half-cell potential is used to characterize the model accuracy. Half-cell potential is the electrochemical term for the negative or positive electrode potential (Section 4.2.4).

2. Formulation of the model

The most common LIB model was developed by Newman et al. [16] and is termed macrohomogeneous model. Considering the complexity of the system, in this paper a simplified PDE system has been developed which can be solved in one time (and length) scale and also on multiple time (or length) scales. In the following the complex geometry of a LIB is presented and later a simplified model is introduced.

2.1. Multi-scale modeling domains of a Li-ion battery

Fig. 1 shows a cylindrical type 18650 Li-ion battery's internal structure reconstructed from X-ray computed tomography data. The battery type refers to its geometrical size, namely a diameter of 18 mm and a length of 65 mm. The cutout shows the macroscopic structure, which consists of the rolled electrode layers around an internal support (altogether called jelly roll), the two terminals (negative and positive), and the current collector inserted in between the layers. The current collector (shown in Fig. 1) collects the electrical current generated by the electrochemical process in the jelly roll and connects it to the negative terminal (positive terminal has the same feature, but that current collector is not visible in Fig. 1). Simultaneously the generated heat flows from inside to outside. Despite the apparent cylindrical symmetry, the layers are not constant in thickness and the current is collected at a particular point of the electrode. This symmetry breaking may not influence considerably the macroscopic behavior, however modeling of manufacturing differences and of safety events, which are triggered by local phenomena (e.g. linked to the creation of hot spots) requires the incorporation of these details of the geometry. While heat transfer occurs at cm scale, the sources of current and heat are the electrodes, whose length scale is ca. 100–200 μm (i.e. 2–3 order of magnitude smaller than the macroscopic scale).

The cross-section of the same battery in Fig. 2 shows the electrode-scale structure. The electrodes are usually prepared by coating the electro-active material on the surface of a thin metal foil. The anode layer (negative electrode) consists of
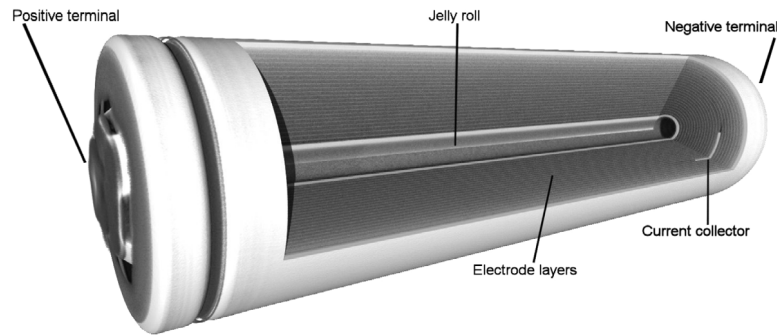
Fig. 1. The macroscopic structure of a cylindrical Li-ion battery reconstructed from X-ray computer tomography data.
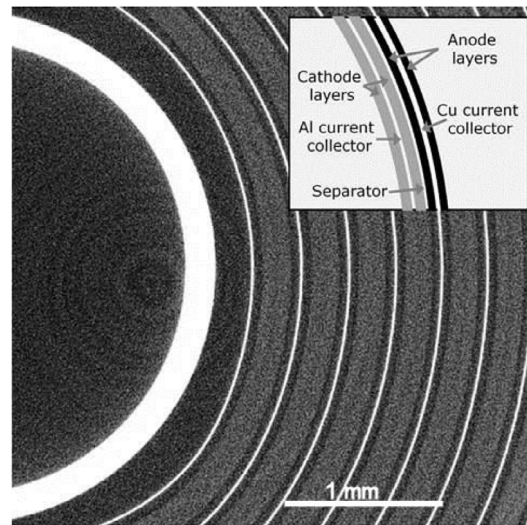

Fig. 2. Cross section of the electrode-scale structure reconstructed from X-ray computed tomography (Fig. 1).

copper foil and a porous graphite layer on both sides. The copper foil is visible as bright line, while the graphite as darker area around it. The cathode layer (positive electrode) is made by aluminum (Al) foil and Lithium iron phosphate (LFP) as an electro-active material. As Al and LFP has similar density and cross section for X-ray absorption, they exhibit similar contrast. In spite of that Al can be seen as little bit darker line in the middle of cathode layer. The separator between the anode and cathode layer is made of polymer and not visible by X-ray, because of its low density. The inset in Fig. 2 shows the schematic representation of the different electrode layers, for the sake of better understanding. The energy is stored in the form of chemical energy of Li-ion inside the electroactive material particles, which are shown in Fig. 3.

The pore-scale structure (Fig. 3) consist of 1–2 magnitudes smaller objects (sub μm length scale), than the electrode scale structure (sub mm). During operation the Li-ions are squeezed out of or pumped into the cathode's electro active particles by the charge transfer processes (charging and discharging) occurring only at the surface (sub nm scale). Then Li-ions shuttle towards or from the anode, passing a few hundred μm in the pores and the separator. Processes at the anode can be described similar to the cathode (above). Simultaneously electrons are transferred towards the Al foil through and external load (e.g. electric engine), closing the circuit.

Common numerical models do not simulate the behavior of a LIB at all scales (from cm to nm), as presented in Figs. 1–3. The porous structure is usually homogenized by different methods [17] and effective transport and kinetic coefficients are assigned to the continuum model. The macroscopic 3D structure of the electrode in the battery is usually simplified into a 1D model by assuming perfect cylindrical symmetry and this approach is also followed in this work. Mathematically the three parts of a LIB electrode, the anode, cathode and the separator can be described similarly. As an example the anode is described in 1D in the following.

Despite of the geometrical simplification, homogenized continuum models can simulate the main physico-chemical characteristics of a LIB. Moreover the developed and presented model can be solved in a splitted and non-splitted way and allows the comparison of different multi-scale modeling algorithm.
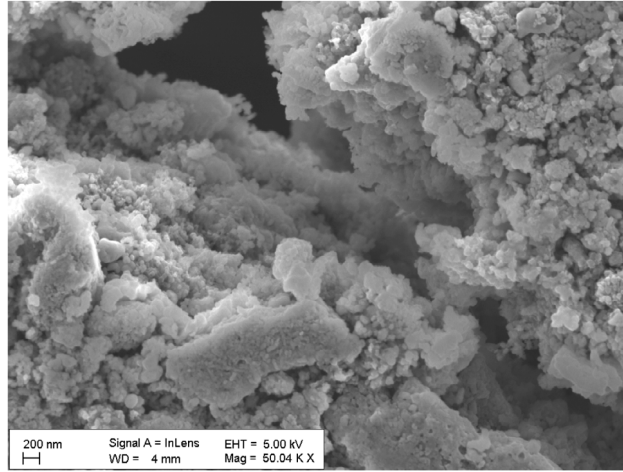
Fig. 3. The pore-scale structure of the cathode layer as imaged by scanning electron microscopy (SEM).

Table 1
Governing variables of a battery cell.

Variables	Description
$\varphi_1(x, t)$	Electric potential in the solid phase
$\varphi_2(x, t)$	Electric potential in the electrolyte
$i_1(x, t)$	Electric current in the solid phase
$i_2(x, t)$	Ionic current in the electrolyte
$N(x, t)$	The material flux of species (Li^+, A^-)
$c(x, t)$	Li^+ ion concentration in the electrolyte
$c_s(x, t)$	Li^+ concentration in the solid phase
$J_1(c_s, c, \eta)$	Source term
$\eta(x, t)$	Overpotential
$I(t)$	Cell current

2.2. Physico-chemical model

In the macrohomogeneous model developed by Newman et al. [16] 2 independent (time t and space x) and 10 dependent variables are defined, which are listed in Table 1. The applied physico-chemical constants and material coefficients are summarized in Table 2. Applying the conservation of charge for currents and conservation of mass for ions (Li-ion and anion) yield the following governing equations:

The potential distribution in the solid phase is

$$-\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial \varphi_1}{\partial x}(x, t)\right) + J_1(c_s, c, \eta) = 0 \tag{4}$$

J_1 is the source term and equals to the sum of charge transfer (electrochemical) and charge accumulation (capacitive) reaction, and can be expressed as

$$J_1 := a_{n,p} \cdot i_0 \cdot K(c_s, c) \cdot E(c_s, \eta) + a_{n,p} \cdot C_{dl}\frac{\partial \eta}{\partial t} \tag{5}$$

where

$$K(c_s, c) := (c_{max} - c_s(x, t))^{\alpha_n}(c_s(x, t))^{\alpha_p}(c(x, t))^{\alpha_p} \tag{6}$$

and

$$E(c_s, \eta) := \exp\left(\frac{\alpha_n F}{RT}\eta(x, t) - \varphi_{st}(c_s(x, t))\right) - \exp\left(-\frac{\alpha_p F}{RT}\eta(x, t) + \varphi_{st}(c_s(x, t))\right) \tag{7}$$

are the electrochemical reaction kinetic equations and

$$\eta = \varphi_1 - \varphi_2 - \varphi_{st}$$

is the overpotential. The potential distribution in the electrolyte phase is

$$-\kappa_{eff}\frac{\partial \varphi_2}{\partial x}(x, t) + \frac{\kappa_{eff}RT}{F} \cdot S(c) - J_1(c_s, c, \eta) = 0 \tag{8}$$

Table 2
List of symbols.

Symbol	Description
σ	Conductivity of solid phase
κ	Conductivity of electrolyte phase
σ_{eff}	Effective conductivity of solid phase
κ_{eff}	Effective conductivity of electrolyte
R	Universal gas constant
T	Temperature
F	Faraday constant
f_A	Activity coefficient of salt
t_+^0	Transference number of Li-ion
v	Stoichiometric number of Li-ion
D	Effective diffusion coefficient of Li$^+$ in the electrolyte
z	Charge number of species
a	Specific surface area
i_0	Exchange current density
c_{max}	Maximum Li$^+$ content in a particle
α	Charge transfer coefficient
C_{dl}	Double layer capacitance
φ_{st}	Standard potential
ε	Porosity
L	Thickness of active layer
n, p, s	In subscript: Negative, Positive, Separator
a, c	In subscript: Anodic, Cathodic

where

$$S(c) := \left[1 + \frac{\partial \ln f_A(c)}{\partial \ln c}(x, t) \right] \cdot (1 - t_+^0) \frac{\partial \ln c}{\partial x}(x, t). \tag{9}$$

The concentration distribution of Li ions in the electrolyte phase is

$$\frac{\partial}{\partial x} \left(D \frac{\partial c}{\partial x}(x, t) \right) + \frac{(1 - t_+^0)}{F} J_1(c_s, c, \eta) = \varepsilon_{n,p} \frac{\partial c}{\partial t}(x, t) \tag{10}$$

and in the solid phase (active material which stores Li) is

$$\frac{\partial c_s}{\partial t}(x, t) = -(1 - \varepsilon) K(c_s, c) E(c_s, \eta). \tag{11}$$

Because the same current flows through the negative and positive terminals the respective boundary conditions for every $t > 0$ are

$$\frac{\partial \varphi_1}{\partial x}(0, t) = -\frac{I}{\sigma_{eff}}, \qquad \frac{\partial \varphi_2}{\partial x}(0, t) - \frac{RT}{F} \cdot S(c(0, t)) = 0 \tag{12}$$

and

$$\frac{\partial \varphi_1}{\partial x}(L, t) = 0, \qquad \frac{\partial \varphi_2}{\partial x}(L, t) - \frac{RT}{F} \cdot S(c(L, t)) = -\frac{I}{\kappa_{eff}}. \tag{13}$$

However there is no material flow from and into the system therefore the Neumann-boundary condition holds for the concentration of Li-ions in the electrolyte:

$$\frac{\partial c}{\partial x} = 0, \quad \text{at } x = [0, L], \ t > 0 \tag{14}$$

and in the solid phase:

$$\frac{\partial c_s}{\partial x} = 0, \quad \text{at } x = [0, L], \ t > 0. \tag{15}$$

The initial conditions are assuming steady state at $t = 0$ for all $0 < x < L$:

$$\varphi_1(x, 0) = \varphi_2(x, 0) = 0, \qquad c(x, 0) = c_0, \qquad c_s(x, 0) = c_{snp} \tag{16}$$

where s, n, p refer to Li-ion concentration in negative and positive electro-active particles (solid phase).

2.3. Derivation of the canonical form

In the following the previously defined equations are simplified to the dimensionless form and the canonical equations. Before transforming (4), (8), (10) and (11) into unified space and time the following simplifications are applied. Let the charge transfer coefficients be equal to 1 ($\alpha_i \equiv 1$), and the activity coefficient of the salt be constant ($f_A \equiv constant$). These simplifications do not alter the physico-chemical model significantly, but make the derivation and comparison of the error between the splitted and non-splitted solution easier as explained below.

The main transformation is to express the potentials (φ_1, φ_2, φ_{st}) by the overpotential η which reduces the number of the equations and results a simpler form of Newman's model. (4) and (8) are amended as follows by using the notations in (5)–(7):

$$-\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}(x,t)\right) + \frac{\partial}{\partial x}\left[\frac{\kappa_{eff}RT}{F}\cdot S(c)\right] = 2\cdot a_{n,p}\cdot i_0\cdot K(c_s,c)\cdot E(c_s,\eta) + a_{n,p}\cdot C_{dl}\frac{\partial\eta}{\partial t}. \tag{17}$$

Next goal is to express φ_2 from (17) by the overpotential η. First, let $\eta^\star$ be defined as follows:

$$\eta^\star(x,t) := \frac{F}{RT}\eta(x,t). \tag{18}$$

By substituting (18) into (17) and after some rearrangement we get the following formula with corresponding initial and boundary conditions:

$$\frac{\partial\eta^\star}{\partial t} = \frac{F}{a_{n,p}C_{dl}RT}\left[-\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x} - \frac{\kappa_{eff}RT}{F}\cdot S(c)\right) - 2\cdot i_0\cdot a_{n,p}\cdot K(c_s,c)\cdot E(c_s,\eta^\star)\right]. \tag{19}$$

Next step is the time transformation of (19) to dimensionless time $t^\star$ defined by

$$t^\star := \frac{t}{\varrho} \quad \text{where } \varrho := \frac{t}{a_{n,p}C_{dl}\left(\frac{1}{\kappa_{eff}} + \frac{1}{\sigma_{eff}}\right)L^2}. \tag{20}$$

Furthermore let $u(x,t^\star)$ be chosen as follows:

$$u(x,t^\star) := \eta^\star(x,\varrho t^\star). \tag{21}$$

By applying the chain rule of multivalued functions with some mathematical rearrangement and then substituting the value of p into (19) we reach the following formula:

$$\frac{\partial u}{\partial t^\star}(x,t^\star) = -\nu^2\cdot K(c_s,c)\cdot E(c_s,u)$$
$$-\frac{F}{RT}\left(\frac{1}{\kappa_{eff}} + \frac{1}{\sigma_{eff}}\right)L^2\cdot\left(\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}(x,\varrho t^\star)\right) - \frac{\partial}{\partial x}\left[\frac{\kappa_{eff}RT}{F}\cdot S(c)\right]\right) \tag{22}$$

where ν^2 is equal to the dimensionless exchange current density, namely:

$$\nu^2 := 2a_{n,p}\left(\frac{1}{\kappa_{eff}} + \frac{1}{\sigma_{eff}}\right)L^2\frac{F}{RT}i_0.$$

Expanding the bracket $\left(\frac{1}{\kappa_{eff}} + \frac{1}{\sigma_{eff}}\right)$ and applying algebraic transformations (the details can be found in the Appendix) (22) results in the following form without φ_2:

$$\frac{\partial u}{\partial\tau}(x,t^\star) = L^2\frac{\partial^2 u}{\partial x^2}(x,t^\star) + L^2\frac{\partial S}{\partial x}(c) - \nu^2\cdot K(c_s,c)\cdot E(c_s,u). \tag{23}$$

The last step is the space transformation, which is based on substitution of the dimensionless distance $X := \frac{x}{L}$ yielding the following form:

$$\frac{\partial U}{\partial\tau}(X,t^\star) = \frac{\partial^2 U}{\partial X^2}(X,t^\star) + \mu\frac{\partial}{\partial X}\left(\frac{1}{C(X,t^\star)}\frac{\partial C}{\partial X}(X,t^\star)\right) - \nu^2\cdot K(C_s,C)\cdot E(C_s,U) \tag{24}$$

where $U(X,t^\star) := u(LX,t^\star)$ and $\mu := (1 - t_+^0)$, furthermore C_s, C and U are dimensionless variables and represent the concentration in the solid phase, the concentration of electrolyte and the overpotential, respectively. The dimensionless forms of c_s and c have been calculated in the same way as the dimensionless over potential U. To prescribe the final form of the PDE-system it is necessary to introduce a few more notations:

$$J_2(C_s,C,U) := K(C_s,C)\cdot E(C_s,U)$$
$$\delta(t^\star) := -L\frac{F}{\sigma_{eff}RT}I(t^\star).$$

Table 3
Dimensionless parameters and their values used in the simplified model.

Property	Symbol	Value
Li$^+$ intercalation ratio	Γ	23.56e−04
Diffusion and migration ratio	ψ	0.25
Charge to mass ratio	θ	94.25e−04
Reaction rate	ν^2	14.41
Current	δ	−1.04 or 0
Conductivity ratio	ξ	95
Linear constants	A, B	4, −7
Initial Li-ion concentration in particle	c_{snp}	0.5
Initial Li-ion concentration in electrolyte	c_0	1
Potential conversion factor	Φ	39.61

Table 4
Properties of a porous anode taken from [16].

Symbol	Unit	Value
σ_{eff}	S m^{-1}	1.9
κ_{eff}	S m^{-1}	0.02
R	J mol^{-1} K^{-1}	8.314
T	K	293
F	C mol^{-1}	96.500
t_+^0		0.5
D	cm^2 s^{-1}	10e−5
a	m^{-1}	100000
i_0	A cm^{-2}	3.6e−6
c_{max}	mol cm^{-3}	0.6
α		1
C_{dl}	F cm^{-2}	0.01
ε	%	0.2
L	cm	0.1

With the appropriate notations and assuming, that $X \in [0, 1]$ and $t^\star > 0$ the final dimensionless form of LIB's system can be prescribed by:

$$\frac{\partial C_s}{\partial \tau}(X, t^\star) = -\Gamma J_2(C_s, C, U), \qquad \frac{\partial C}{\partial t^\star}(X, t^\star) = \psi \frac{\partial^2 C}{\partial X^2}(X, t^\star) + \theta J_2(C_s, C, U)$$

$$\frac{\partial U}{\partial t^\star}(X, t^\star) = \frac{\partial^2 U}{\partial X^2}(X, t^\star) + \mu \frac{\partial}{\partial X}\left(\frac{1}{C(X, t^\star)}\frac{\partial C}{\partial X}(X, t^\star)\right) - \nu^2 J_2(C_s, C, U). \tag{25}$$

The standard potential of the electrode φ_{st} is a thermodynamic property of the active material and depends on C_s. However, it can be approximated in most of the cases by a linear relationship in the following form

$$\varphi_{st}(C_s) = A + B \cdot C_s \tag{26}$$

where A and B are arbitrary constants. Consequently the corresponding initial and boundary conditions are:

$$C_s(X, 0) = c_{snp}, \qquad C(X, 0) = c_0, \qquad U(X, 0) = \Phi(A + Bc_{snp}) \tag{27}$$

where Φ is the conversion factor to transform the standard potential φ_{st} to dimensionless potential.

$$\frac{\partial C_s}{\partial X}(0, t^\star) = \frac{\partial C_s}{\partial X}(1, t^\star) = 0, \qquad \frac{\partial C}{\partial X}(0, t^\star) = \frac{\partial C}{\partial X}(1, t^\star) = 0$$

$$\frac{\partial U}{\partial X}(0, t^\star) = \delta(t^\star), \qquad \frac{\partial U}{\partial X}(1, t^\star) = -\delta(t^\star)\xi \tag{28}$$

The used dimensionless parameters in (25), (27) and (28) and their meaning are summarized in Table 3. The values are calculated by using the input parameters in Table 4.

2.4. Numerical model

2.4.1. Splitting strategy

As it has been shown, the mathematical model of LIB can be transformed into the PDE (25) with appropriate initial (27) and boundary (28) conditions. For the sake of better readability the dimensionless space (X) and the dimensionless time ($t^\star$) will be denoted by x and t, respectively in the rest of the paper.

In order to reach an operator form of (25), (27) and (28), the following notations are introduced:

$$\vec{w}(x, t) := [C_s \, C \, U]^T(x, t), \qquad \vec{w}(x, 0) = \vec{w}_0(x) := \begin{bmatrix} c_{snp} \\ c_0 \\ \Phi(A + Bc_{snp}) \end{bmatrix}$$

$$\frac{\partial \vec{w}}{\partial x}(0, t) = g_1(t) := \begin{bmatrix} 0 \\ 0 \\ \delta(t) \end{bmatrix}, \qquad \frac{\partial \vec{w}}{\partial x}(1, t) = g_2(t) := \begin{bmatrix} 0 \\ 0 \\ -\delta(t)\xi \end{bmatrix}$$

$$(M(\vec{w}))(x, t) := \begin{bmatrix} 0 & 0 & 0 \\ 0 & \psi \dfrac{\partial^2}{\partial x^2}C & 0 \\ 0 & 0 & \dfrac{\partial^2}{\partial x^2}U \end{bmatrix}(x, t)$$

$$(f_0(\vec{w}))(x, t) := \begin{bmatrix} -\Gamma J_2(C_s, C, U) \\ \theta J_2(C_s, C, U) \\ \mu \cdot \dfrac{\partial}{\partial x}\left(\dfrac{1}{C}\dfrac{\partial C}{\partial x}\right) - v^2 J_2(C_s, C, U) \end{bmatrix}(x, t) \qquad (29)$$

where

$(M(\vec{w}))(x, t)$ is called the diffusion operator   (30)

$(f_0(\vec{w}))(x, t)$ is called the source operator.   (31)

Based on the notations in (29) the operator form of PDE-system (25), (27) and (28) can be defined as follows:

$$\frac{\partial \vec{w}}{\partial t}(x, t) = (M(\vec{w}))(x, t) + (f_0(\vec{w}))(x, t)$$

$$\vec{w}(x, 0) = \vec{w}_0, \qquad \frac{\partial \vec{w}}{\partial x}(0, t) = g_1(t), \qquad \frac{\partial \vec{w}}{\partial x}(1, t) = g_2(t). \qquad (32)$$

The problem (32) can be solved by many ways, therefore it is needed to determine

1. The sub-problems (splitting)
2. The splitting algorithm
3. The solution order of the sub-problems and
4. The numerical solution scheme for each sub-problem

The main benefit of applying operator splitting methods for battery simulation is that one of the operators can be solved at macroscopic scale, while the other one can be solved at microscopic scale. The microscopic processes embedded in the macroscopic process can be solved on a finer mesh, which may increase accuracy without a considerable increase of running time. Usually the operator with more complicated mathematical structure is chosen to be solved on a finer mesh. However we investigated all possible combinations of the above points. After testing all of the feasible splitting constructions, the most effective choice appeared to be solving the source operator on a finer mesh, because the diffusion operator is linear, while the source operator is non-linear. This selection also follows the physical processes and the general approach of multi-scale models, according to Section 2. In the following pore scale is called microscale, while electrode scale is called macro scale.

The selection of solution order including the merging of the sub-solutions of the two sub-problems and the discretization scheme are less trivial, though. To study the effect of splitting algorithm and solution order the following cases are analyzed regarding accuracy and running time:

- Sequential splitting
- Reverse sequential splitting
- Strang–Marchuk splitting
- Reverse Strang–Marchuk splitting
- Symmetrical splitting

Another advantage of the operator splitting technique is that the different sub-problems can be solved by different numerical methods. In this work the following numerical schemes are applied:

Full explicit: The non-splitted problem is solved by explicit Euler scheme
Splitted explicit: Both of the splitted sub-problems are solved by explicit Euler scheme
Splitted IMEX: A combination of explicit and implicit methods is applied in such a way, that the diffusion operator is discretized implicitly and the source operator explicitly [18].

Numerical algorithms were developed to solve the system implicitly also, but the convergence of the method was not satisfactory. The full implicit scheme has several numerical difficulties because of the non-linear source term, which results in slow and computational intensive calculations. For these reasons the full implicit scheme was not compared in this paper.

2.4.2. Discretization

For the numerical solution of problem (32) the following mesh is defined for the macroscopic and microscopic problem, respectively. First, an appropriate grid is generated for the macroscopic problem. Let $\omega_{h,\tau}^{mac}$ be a mesh, which consists of the (x_i, t_k) mesh-points, where h and τ denote the chosen spatial and time resolution of the mesh, according to the following:

$$x_i = ih, \quad h = \frac{1}{N_x} \quad i = 0, 1, 2, \ldots, N_x$$

$$t_k = k\tau, \quad \tau = \frac{T}{N_T} \quad k = 0, 1, 2, \ldots, N_T \tag{33}$$

where N_x and N_T mark the numbers of division parts in space and time. Then we introduce a finer mesh for the microscopic problem. Let this mesh denoted by $\omega_{h,\Delta\tau}^{mic}$ which consists of the (x_i, t_n) mesh-points, where h and $\Delta\tau$ denote the chosen spatial and time resolution, respectively. In this case:

$$x_i = ih, \quad h = \frac{1}{N_x} \quad i = 0, 1, 2, \ldots, N_x$$

$$t_n = n\Delta\tau, \quad \Delta\tau = \frac{\tau}{N_\tau} \quad n = 0, 1, 2, \ldots, (N_T \cdot N_\tau) \tag{34}$$

where N_τ marks the number of subdivision parts in time and space. Let us perceive that $(N_T \cdot N_\tau) \cdot \Delta\tau = N_T \cdot \tau = T$ which means that the time interval is the same as by mesh $\omega_{h,\tau}^{mac}$ only with finer time steps. There are two things worth mentioning: First, $\omega_{h,\Delta\tau}^{mic}$ contains every point from mesh $\omega_{h,\tau}^{mac}$ and additionally extra points. The two sets of mesh-points are not necessarily needed to overlap, but this alternative is not investigated here. Second, spatial resolution of the mesh is not changed, because the convergence criteria for FDM is linked with time through $\frac{\Delta\tau}{h^2}$. Therefore varying both of the independent variables would make the comparison of the accuracy of different algorithms more difficult. Varying spatial mesh however is necessary, if an embedded partial differential system is solved for the source term.

Hereafter the introduction of a corresponding vector space $\Xi(\omega_{h,\Delta\tau}^{mic})$ is needed, where the approximated mesh-functions are interpreted on $\omega_{h,\Delta\tau}^{mic}$ (defined in (34)). Our aim is to find a series of mesh-functions $\left(y_i^n\right)_j := \left(y_{h,\Delta\tau}\right)_j (x_i, t_n) \in \Xi(\omega_{h,\Delta\tau}^{mic})$ which approximates well the jth components of vector function $(\vec{w})_j (x_i, t_n)$ defined in (29) at the mesh-point $(x_i, t_n) \in \omega_{h,\Delta\tau}^{mic}$. Let us denote $j = 1, 2, 3$ the components of the solution C_s, C and U, respectively.

The *fully explicit* method to the problem (32) with the corresponding initial and boundary conditions is for $j = 1, 2, 3$

$$\frac{(y_i^n)_j - (y_i^{n-1})_j}{\tau} = \Upsilon_j \cdot \frac{(y_{i+1}^{n-1})_j - 2(y_i^{n-1})_j + (y_{i-1}^{n-1})_j}{h^2} + \left(f_0\left(y_i^{n-1}\right)\right)_j \tag{35}$$

where Υ_j denotes the appropriate coefficient inside the diffusion operator, i.e., $\Upsilon_1 = 0$, $\Upsilon_2 = \psi$ and $\Upsilon_3 = 1$ and $\left(f_0\left(y_i^{n-1}\right)\right)_j$ denotes the jth component of source term operator defined in (34).

In the splitted explicit method the whole system is divided into two sub-problems according to (32), which yields the following discretized equation system for the diffusion operator for $j = 1, 2, 3$

$$\frac{(y_i^n)_j - (y_i^{n-1})_j}{\tau} = \Upsilon_j \cdot \frac{(y_{i+1}^{n-1})_j - 2(y_i^{n-1})_j + (y_{i-1}^{n-1})_j}{h^2} \tag{36}$$

and for the source operator for $j = 1, 2, 3$

$$\frac{(y_i^n)_j - (y_i^{n-1})_j}{\Delta\tau} = \left(f_0\left(y_i^{n-1}\right)\right)_j. \tag{37}$$

The discretization of splitted IMEX numerical scheme is similar to the previous case's diffusion and source term operators. However (36) is approximated by an implicit approach and the non-linear source term is solved by the combination of the explicit Euler and the single-step Newton iteration methods. Using the notations introduced in (35), the following formula has been developed for the diffusion operator for $j = 1, 2, 3$

$$\frac{(y_i^n)_j - (y_i^{n-1})_j}{\tau} = \Upsilon_j \cdot \frac{(y_{i+1}^n)_j - 2(y_i^n)_j + (y_{i-1}^n)_j}{h^2} \tag{38}$$

and for the source operator for $j = 1, 2, 3$

$$\frac{(y_i^n)_j - (y_i^{n-1})_j}{\Delta\tau} = -\frac{\left(f_0\left(y_i^{n-1}\right)\right)_j}{\left(f_0'\left(y_i^{n-1}\right)\right)_j} \tag{39}$$

where $\left(f_0'\left(y_i^{n-1}\right)\right)_j$ denotes the jth components of source term operator's derivative.
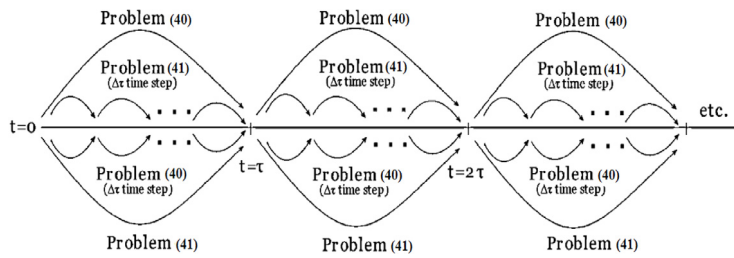
Fig. 4. The flow chart of symmetrical splitting algorithm.

2.5. The algorithm of operator splitting schemes for Lithium-ion batteries

In this section the calculation of the approximate solution from the results of the two sub-problems are described. In all cases, the source operator is solved on a finer mesh by using splitted explicit Euler or IMEX methods presented in Section 2.4.2.

2.5.1. Sequential and reverse sequential splitting

The sequential splitting solves the two sub-problems consecutively and uses the previous step's solution as the initial condition of the next step. The forward sequential splitting scheme solves the diffusion operator first, and consecutively the source operator on a finer mesh:

Problem 1. Diffusion operator

$$\frac{\partial \vec{w}_1^{(1)}}{\partial t}(x, t) = \left(M \left(\vec{w}_1^{(1)} \right) \right)(x, t), \quad 0 < t \leq \tau$$
$$\vec{w}_1^{(1)}(x, 0) = \vec{w}_0(x), \quad x \in (0, 1)$$
$$\frac{\partial \vec{w}_1^{(1)}}{\partial x}(0, t) = g_1(t), \qquad \frac{\partial \vec{w}_1^{(1)}}{\partial x}(1, t) = g_2(t), \quad 0 < t < \tau. \tag{40}$$

Problem 2. Source operator (on a finer mesh)

$$\frac{\partial \vec{w}_2^{(1)}}{\partial t}(x, t) = \left(f_0 \left(\vec{w}_2^{(1)} \right) \right)(x, t), \quad 0 < t \leq \tau$$
$$\vec{w}_2^{(1)}(x, 0) = \vec{w}_1^{(1)}(x, \tau), \quad x \in (0, 1) \tag{41}$$

where the subscripts of $\vec{w}$ correspond to the solution of each sub-problem and the superscript is the splitting step. Furthermore Problem 2 (41) is solved independently N_τ times to reach the solution in point τ, because $N_\tau \cdot \Delta \tau = \tau$.

The reverse order is defined when the source operator is solved first N_τ times (on a finer mesh) and consecutively the diffusion operator.

2.5.2. Symmetrical splitting

This splitting method combines the simple sequential and the reverse sequential splitting methods. The main idea of the symmetrical splitting is that the sequential splitting solutions are calculated in both directions, and then their average value is taken as a solution. The diffusion operator and source operator are the same as in Problems (40) and (41), respectively. In Fig. 4 the procedure of symmetrical splitting algorithm is presented. The solution's algorithm is the following:

1. Solving Problem (41) on the time interval $0 < t < \tau$, with arbitrary $\Delta \tau$ time step N_τ times independently to reach point τ, where the initial conditions are given in (32).
2. Solving Problem (40) on the same time domain using τ time step, where the initial conditions are defined by the solution of Problem (41).
3. Solving Problem (40) on the same time interval, with τ time step, where the initial conditions are given in (32).
4. Solving Problem (41) on the same time domain, with arbitrary $\Delta \tau$ time step N_τ times independently to reach point τ, where the initial conditions are defined by the solution of Problem (40).
5. The average of Problem (41) and Problem (40) is the solution of the whole problem.
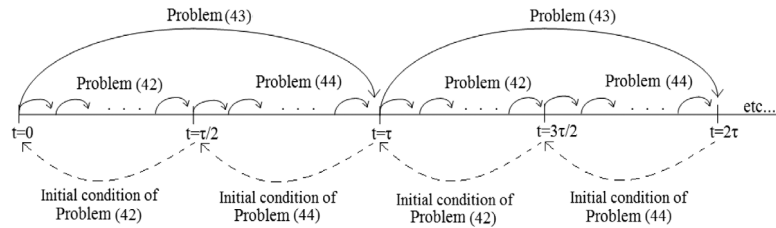6. Repeat from step 1 by using the appropriate time domain.

Fig. 5. The flow chart of Strang–Marchuk splitting scheme.

2.5.3. Strang–Marchuk splitting

In the forward Strang–Marchuk splitting (S–M) scheme the diffusion operator (macroscopic problem) is solved on different halves of the $(0, \tau)$ time interval and the microscopic problem is solved on the whole $(0, \tau)$ interval on the finer mesh. Each problem is solved by using an adequate numerical method with properly chosen time steps. The reverse S–M scheme solves the problems in reverse order, namely the microscopic problem is solved two times in the different halves of the $(0, \tau)$ interval and the macroscopic problem passes along on the whole $(0, \tau)$ time interval. Consequently S–M consists of three problems and the reverse S–M splitting algorithm is as follows:

Problem 1. Source operator (on a finer mesh)

$$\frac{\partial \vec{w}_1^{(1)}}{\partial t}(x, t) = \left(f_0 \left(\vec{w}_1^{(1)} \right) \right)(x, t), \quad 0 < t \leq \frac{\tau}{2}$$

$$\vec{w}_1^{(1)}(x, 0) = \vec{w}_0(x), \quad x \in (0, 1). \tag{42}$$

Problem 2. Diffusion operator

$$\frac{\partial \vec{w}_2^{(1)}}{\partial t}(x, t) = \left(M \left(\vec{w}_2^{(1)} \right) \right)(x, t), \quad 0 < t \leq \tau$$

$$\vec{w}_2^{(1)}(x, 0) = \vec{w}_1^{(1)}(x, \tau/2), \quad x \in (0, 1)$$

$$\frac{\partial \vec{w}_2^{(1)}}{\partial x}(0, t) = g_1(t), \qquad \frac{\partial \vec{w}_2^{(1)}}{\partial x}(1, t) = g_2(t), \quad 0 < t < \tau. \tag{43}$$

Problem 3. Source operator (on a finer mesh)

$$\frac{\partial \vec{w}_3^{(1)}}{\partial t}(x, t) = \left(f_0 \left(\vec{w}_3^{(1)} \right) \right)(x, t), \quad \frac{\tau}{2} < t \leq \tau$$

$$\vec{w}_3^{(1)}(x, 0) = \vec{w}_2^{(1)}(\tau), \quad x \in (0, 1). \tag{44}$$

The superscript denotes the time step of the numerical model and the subscript refers to the number of the splitted sub-problem. Fig. 5 represents the procedure of the above detailed splitting algorithm. The sequence of the Strang–Marchuk splitting algorithm is the following:

1. Solving Problem (42) on the first half of the time interval, i.e., $0 < t < \frac{\tau}{2}$, with arbitrary $\Delta \tau$ time step $N_\tau/2$ times independently to reach point $\tau/2$, where the initial conditions are given in (32).
2. Solving Problem (43) on the whole time sub-domain, i.e., on $0 < t < \tau$ by using τ time step, where the initial conditions are defined by the solution of Problem (42).
3. Solving Problem (44) on the second half of the time interval, i.e., $\frac{\tau}{2} < t < \tau$, with arbitrary $\Delta \tau$ time step $N_\tau/2$ times independently to reach point τ, where the initial conditions are defined by the solution of Problem (43).
4. The solution of Problem (44) is the solution of the whole problem, namely $w_3^{(1)}(\tau) = w_{sp}(\tau)$.
5. Repeat from step 1 by using the appropriate time domain.

2.6. Error calculation

The reference non-splitted solution of the PDE-system in (25), (27) and (28) is calculated by Matlab's built in PDE-system solver, namely "pdepe", which is based on finite element method [19]. Small time step (10^{-5}) and fine mesh were used in order to calculate the reference solution. The numerical solutions of splitting algorithms (denoted by the approximated solution of C_s, C and U where the jth component of $\left(y_i^{(n)} \right)_j := (y)_j (x_i, t_n)$ mesh vector function and $j = 1, 2, 3$) by using

an appropriate numerical scheme, i.e., (36), (37) or (38), (39) and the jth component of reference non-splitted solution $\left(\vec{w}_i^{(n)}\right)_j := (\vec{w})_j(x_i, t_n)$ are compared in different norms for $j = 1, 2, 3$. Using these notations, the relative error matrix of the numerical schemes is expressed by the following for $j = 1, 2, 3$:

$$(A)_j := (Err(x_i, t_n))_j = \frac{\left(\vec{w}_i^{(n)}\right)_j - \left(y_i^{(n)}\right)_j}{\left(\vec{w}_i^{(n)}\right)_j}. \tag{45}$$

The measure of these matrices can be calculated by different norms. In this work the modified l_2 norm is used to characterize the absolute accuracy of the numerical algorithm. To make the solution of the different approximations comparable (independent of the number of the mesh points) the absolute error of the solution ($0 \leq x \leq 1, 0 \leq t \leq T$) is normalized by the number of the mesh points, which results in the average relative error of the splitting in space and time for $j = 1, 2, 3$:

$$\left(\|A\|_{l_2}\right)_j^2 = \frac{\sum_{i=1}^{N_x+1} \sum_{n=1}^{N_T+1} \left|(a_{i,n})_j\right|^2}{(N_x + 1)(N_T + 1)}. \tag{46}$$

This norm can be defined as an analogy of the classical Frobenius matrix norm as

$$(Err)_j = \sqrt{\left(\|A\|_{l_2}\right)_j^2} = \sqrt{\frac{\sum_{i=1}^{N_x+1} \sum_{n=1}^{N_T+1} \left|(a_{i,n})_j\right|^2}{(N_x + 1)(N_T + 1)}}. \tag{47}$$

The overall approximation accuracy of functions C_s, C and U are computed by their arithmetical average, i.e.,

$$TotalErr = \frac{(Err)_1 + (Err)_2 + (Err)_3}{3}. \tag{48}$$

3. Experimental

The models were implemented in MATLAB (version 2012b) and the codes were executed on Higgs, the high performance computational cluster of JRC-IET. Higgs is based on Linux x86-64 architecture and comprises 27 compute nodes, interconnected via Infiniband QDR (quad data rate bus). Each compute node has 16 compute cores (Intel Xeon E5-2670 processors) and 128GB of memory. The graphical representations were also implemented in MATLAB and executed on a standard personal computer (PC).

A Nanotom S X-ray computed tomography system (GE Sensing & Inspection Technologies, phoenix X-ray, Wunstorf, Germany) was used for the investigation of unmodified A123-APR18650M1 Lithium Iron-Phosphate cell. The system is equipped with and X-ray tube with a maximum output power of 15 W and a maximum high voltage of 180 kV in combination with a 2D detector with a dynamic range of 850:1 which consists of 2300 × 2300 pixels. An X-ray energy of 100 kV was selected and the chosen voxel size was in the range of 2.5 to 30 μm (the latter allowing for imaging of the complete cell). 2160 projections were imaged for each dataset using the full pixel resolution of the detector. Suitable software – VG Studio MAX (Volume Graphics, Heidelberg, Germany) – was applied for data visualization and analysis.

The electrochemical measurements were made in an optically transparent cell by EL-CELL. A graphite–Li electrode were placed side-by-side towards the optically transparent window. 1M LiPF6 solution in EMC:DC:DEC (1:1:1) electrolyte was injected into the cell. A PAR 273 potentiostat was used to charge by 20 μA for 10 h until 0V end of charge potential. Digital imaging was performed using a Leica MC120HD camera coupled to an Aristomet Leica microscope to reveal state of charge distribution.

4. Results and discussion

4.1. Solution of the PDE system

The spatial distribution of the dimensionless variables C, C_s and U are shown in Figs. 6–9 at different times solved by a built in function of Matlab during discharge of the negative electrode (graphite). The (dimensionless) discharge current (see in Table 3) is applied from dimensionless time 0 to 4.5 and then the current is switched off, i.e., the system is relaxed. The arrows show the variation of solution in time during constant (blue) and zero (red) currents.

Fig. 6 shows the dimensionless Li-ion concentration inside the active material (state of charge, SOC). During non-zero external current (blue curve and arrow) Li-ion is extracted from the whole electrode, especially from the right side. During relaxation (red curves and arrows) the concentration homogenizes, i.e., Li-ion inserts into the right side from the left side of the electrode (indicated by the opposite direction of the red arrows). It is noticeable that the distribution is
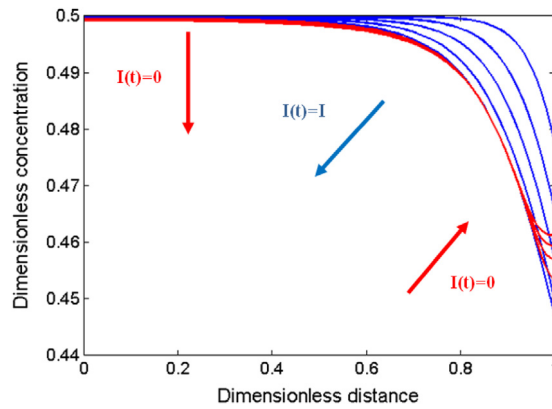
Fig. 6. The variation of Li$^+$ concentration in the particles (C_s) during discharge of the negative electrode.
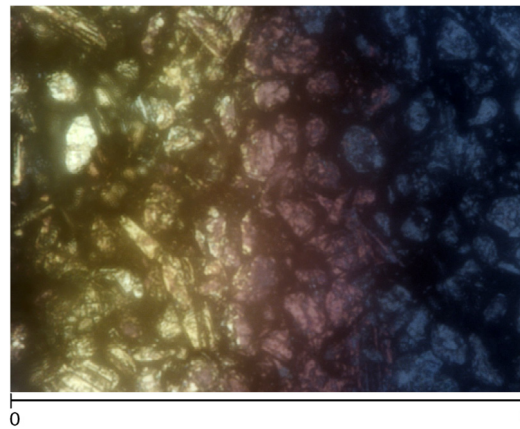

Fig. 7. Optical image of the battery electrode during discharging of the negative electrode. Blue, red and gold color mean approximately 50%, 80%, 100% Li content, respectively.

non-homogeneous because of the difference in the conductivity between the solid and electrolyte phase and the relatively high dimensionless exchange current density (ν^2). The driving force of equilibration is the dependence of standard potential φ_{st} on C_s. If the dependence of the reference potential on C_s is negligible the equilibration does not happen.

Fig. 8 shows the dimensionless overpotential, which is the main driving force of the electrochemical reaction. It shows a highly non-uniform distribution, which indicates a non-homogeneous reaction rate distribution (not seen in figures). After the current is switched off, the dimensionless overpotential stabilizes at a higher value because of the dependence of reference potential (φ_{st}) on the solid Li-ion concentration.

Fig. 9 shows the dimensionless Li-ion concentration in the electrolyte. When the current is cut down, the diffusion of Li-ion in the electrolyte phase equates its concentration, C. This diffusion corresponds to the transport of excess Li-ion from the left part (Fig. 6) to the right during the equilibration of C_s. At the end of discharge C reaches a higher value than initially observed, because Li-ions are extracted (de-intercalated) from the electrode during the discharge process.

The simulated results (Fig. 6) show a qualitative agreement with the measured behavior in Fig. 7. The different colors in Fig. 7 show a highly non-uniform Li-ion distribution, such as the simulated results of C_s in Fig. 6. Blue, red and gold color mean approximately 50%, 80%, 100% Li content, respectively [20]. During relaxation (not seen in Fig. 7) the color difference diminishes and the electrode turns to the same color (i.e., blue).

4.2. Analysis of different numerical schemes

In the next subsections the accuracy and running time of splitting schemes are analyzed.

1. splitted and non-splitted solution
2. different numerical schemes applied for the different splitted sub-problems (operator),
3. five different splitting methods (sequential, reverse sequential, Strang–Marchuk, Strang–Marchuk reverse and symmetrical)
4. measurable cell potential of non-splitted reference solution, sequential, Strang–Marchuk and symmetrical splitting.
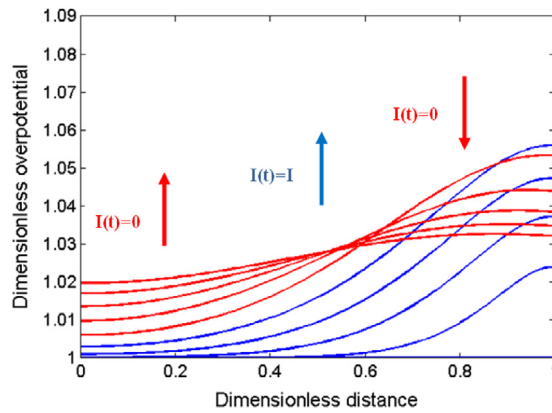
Fig. 8. The variation of the concentration of Li$^+$ ions in the electrolyte phase (C) during discharge of the negative electrode.
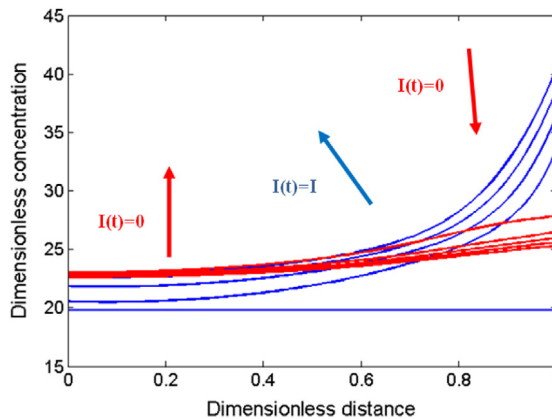

Fig. 9. The variation of dimensionless overpotential (U) during discharge of the negative electrode.
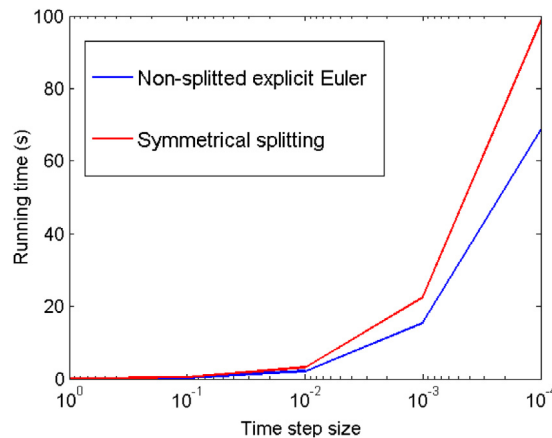

Fig. 10. The running time at different time step size measured for symmetrical splitting and non-splitted explicit-Euler scheme.

4.2.1. Comparison of splitted and non-splitted solution

The operator form (32) of the problem was solved by using the formula (35) for the non-splitted solution and the formula (36)–(37) for the splitted solution. The symmetrical splitting algorithm (detailed in Section 2.5.2) was applied to obtain the splitted solution with a splitting subdivision 10. The errors in both cases were calculated according to (48). The running times were calculated by using the Matlab's *tic tac* function and the results are plotted on Fig. 10 at decreasing time step size (macro time step). The symmetrical splitting solution was obtained without parallel computation techniques.

The running time increases with approximately the same pace as the time step increases for both cases (Fig. 10). In spite of that the source operator in the splitted scheme was solved 10 times more often than in the non-splitted (explicit) scheme,
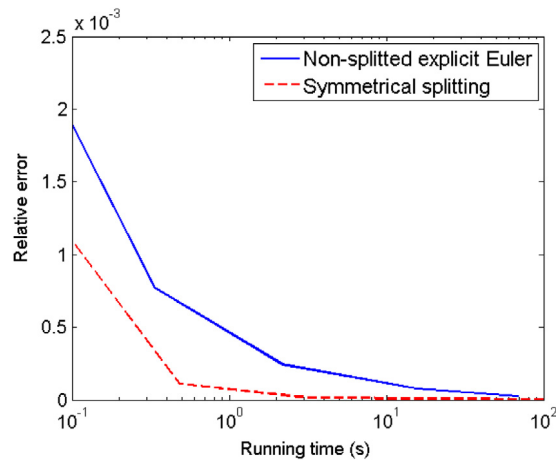
Fig. 11. Comparison of the running time and relative error between the symmetrical splitting and the non-splitted explicit-Euler scheme.

the running time is not increased by 10 times, but approximately 2 times. In the non-splitted solution a complex matrix equation of a non-homogeneous partial differential equation is solved. The computation of the inverse matrix needs more steps, hence more time, than the calculation of the inverse matrix of a homogeneous partial differential equation used in the splitted solution. It is noted, that at very long time step size (> 0.1) the splitted solution was not convergent. In both cases the error decreases with decreasing time step size, except at very small step sizes ($< 10^{-4}$) in the case of the splitted solution. The relative average error of the splitted solution calculated by (48) decreased by approximately 2–5 times compared to the non-splitted explicit Euler at the same running (Fig. 11). The results in Fig. 10 indicate that the running time of the non-splitted solution at the same accuracy is ca. 5–10 times shorter than that of the non splitted solution. Consequently it is worth splitting the LIB system to two simpler problems.

It is important to note that the running time of symmetrical splitting can be reduced further by applying parallel computational technique, because the solutions of the forward and reverse sub-problems inside the algorithm can be solved independently. In this manner the total running time might be apparently halved. In regard to multi-scale modeling the splitting of LIB's solution to macroscopic and microscopic problem does not degrade the accuracy of the solution and not increase the running time. Consequently multi-scale models can be as accurate as a single scale numerical model.

4.2.2. Comparison of splitted explicit and IMEX schemes

One of the main advantages of operator splitting is that it allows the application of different numerical schemes for the source and for the linear diffusion operator. In this section the explicit and IMEX methods are compared. The diffusion operator is solved by explicit (36) or implicit (38) scheme and the source operator is solved by explicit Euler combined with simple iteration (37) or Newton iteration (39). All of the algorithms were tested by using the parameters listed in Tables 3 and 4 with macroscopic time step of 10^{-3}. The microscopic time step size of the source operator is varied between 10^{-3} and 10^{-5} (subdivision is varied between 1 and 100).

Figs. 12–14 show the comparison of the relative average error of C_s, C and U, in Frobenius-norm as a function of the splitting subdivision by using the IMEX methods (blue dotted and red dashed lines) and explicit method (green continuous line) with symmetrical splitting scheme, respectively. Changing the numerical scheme of the macro problem (diffusion operator) from explicit (green continuous) to implicit method (blue dotted) increases the accuracy of C_s and U and does not alter in the case of C. Blue dotted and red dashed lines in Figs. 12–14 show the change in accuracy when explicit Euler (blue dotted) switched to Newton iteration (red dashed). In the case of the micro problem, all schemes equally well. The algorithm of simple iteration (explicit Euler) scheme calculates C_s (Fig. 12) and U (Fig. 14) more accurately but Newton iteration gives smaller error for C (Fig. 13). While the error always decreases in the case of Newton method, the error of U (Fig. 14) increases with the splitting sub-division in the case of the application of simple iteration.

4.2.3. Comparison of different splitting methods

The comparison of the relative error of the explicit and IMEX schemes shows, that the IMEX scheme's accuracy is higher. Therefore, in the following only the IMEX method combined with Newton iteration is examined further. Figs. 15–17 show the splitting error of C_s, C and U, respectively in Frobenius norm as a function of the splitted subdivision for 5 different splitting methods, namely the sequential, reverse sequential, symmetrical, Strang–Marchuk and reverse Strang–Marchuk splitting methods. The macroscopic time step size is 10^{-3}, while the splitting subdivision is varied between 1 and 100.

It can be concluded, that the applied splitting methods performs equally well. Reverse S–M splitting gives the best accuracy for C_s (Fig. 15) and U (Fig. 17), but in the same time it is the less accurate in the case of C (Fig. 16). In Figs. 15–17 it can be also clearly seen that it is not worth to split the time scale at microscale more than 1/10 of macro time scale (shown
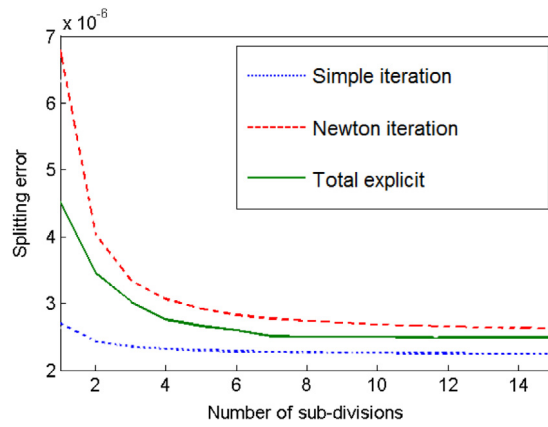
Fig. 12. The error of C_s in Frobenius norm solved by symmetrical splitting method. The diffusion operator was solved by implicit scheme while the source operator with explicit Euler combined with simple (blue dotted line) and Newton iteration (red dashed line). The green continuous line represents the total explicit scheme. The macroscopic time step size is 10^{-3}.
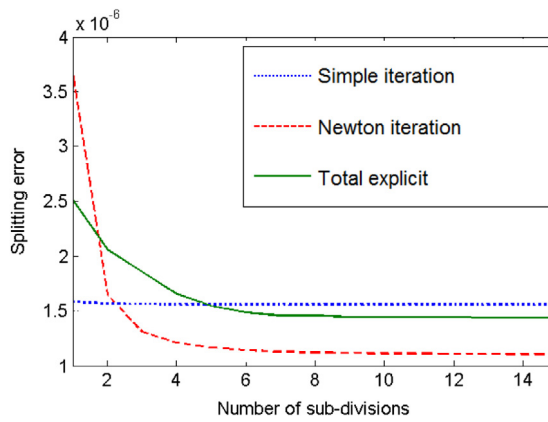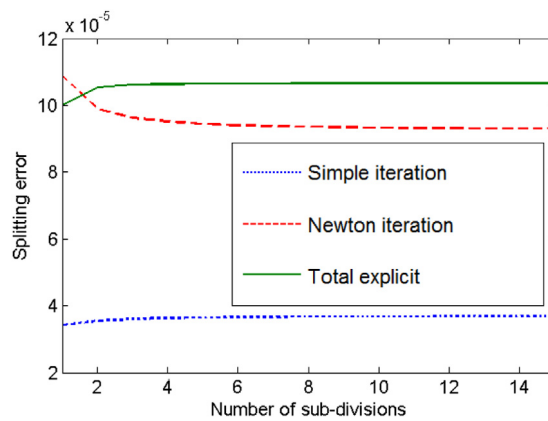

Fig. 13. The error of C in Frobenius norm solved by symmetrical splitting method. The diffusion operator was solved by implicit scheme while the source operator with explicit Euler combined with simple (blue dotted line) and Newton iteration (red dashed line). The green continuous line represents the total explicit scheme. The macroscopic time step size is 10^{-3}.

Fig. 14. The error of U in Frobenius norm solved by symmetrical splitting method. The diffusion operator was solved by implicit scheme while the source operator with explicit Euler combined with simple (blue dotted line) and Newton iteration (red dashed line). The green continuous line represents the total explicit scheme. The macroscopic time step size is 10^{-3}.

by a vertical line in each sub-figure), because the better accuracy expected at smaller time step sizes is probably deteriorated by the increased computational and rounding errors.
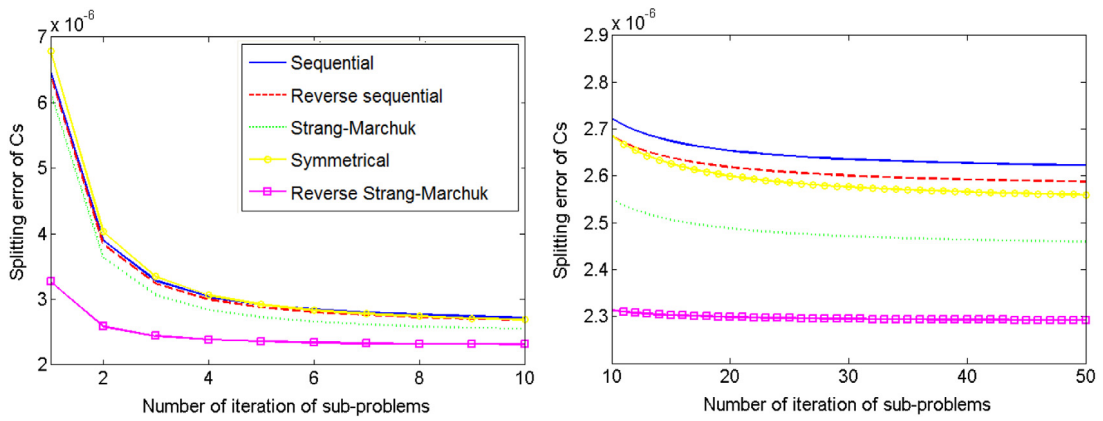
Fig. 15. Accuracy of C_s calculating by 5 different splitting methods. Parameters in Tables 3 and 4 are used with 10^{-3} macroscopic time step size while the microscopic time step size varied between 10^{-3} and 10^{-5}.
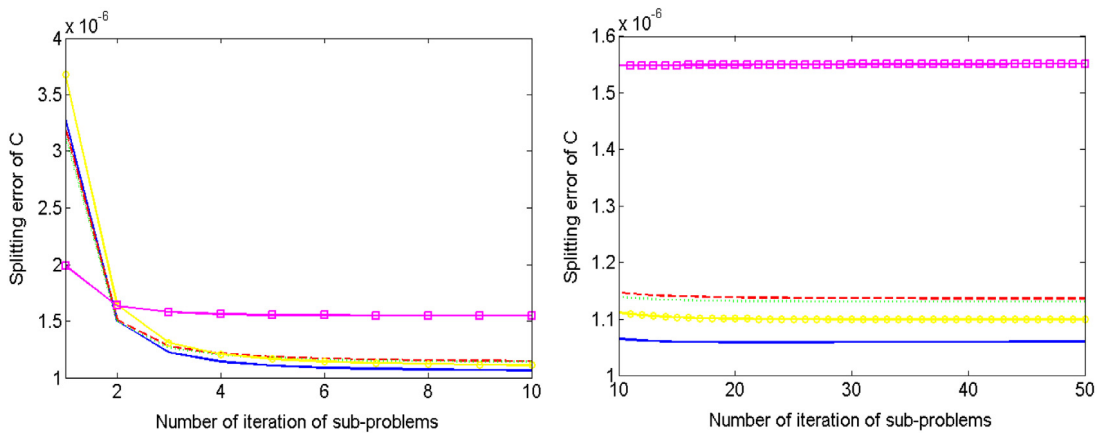

Fig. 16. Accuracy of C calculating by 5 different splitting methods. Parameters in Tables 3 and 4 are used with 10^{-3} macroscopic time step size while the microscopic time step size varied between 10^{-3} and 10^{-5}.
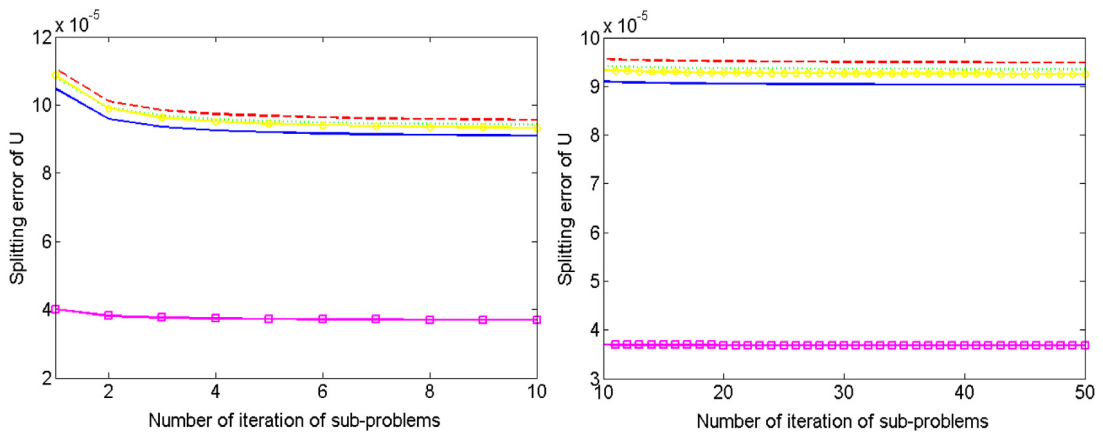

Fig. 17. Accuracy of U calculating by 5 different splitting methods. Parameters in Tables 3 and 4 are used with 10^{-3} macroscopic time step size while the microscopic time step size varied between 10^{-3} and 10^{-5}.

4.2.4. Accuracy of half-cell potential

The convergence and accuracy of a numerical model can be evaluated mathematically in an adequately chosen norm. In practice, the model is validated by the half-cell potential difference V through the negative electrode, namely the difference of $\varphi_1(0, t) - \varphi_2(L, t)$ [21]. In this chapter we calculate this measurable quantity and its accuracy by solving
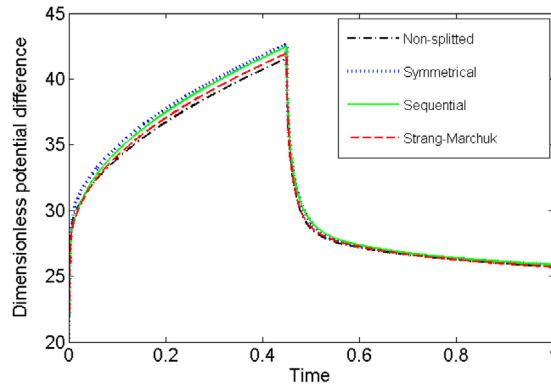
Fig. 18. The dimensionless half-cell potential of the negative electrode.

the system with different splitting algorithms. Rearranging the definition of overpotential we reach $\varphi_1(x, t) = \varphi_2(x, t) + \varphi_{st}(x, t) + \eta(x, t)$. The half-cell potential is then

$$V = \varphi_1(0, t) - \varphi_1(L, t) + \varphi_{st}(L, t) + \eta(L, t). \tag{49}$$

Using (4) and (5) we obtain

$$\sigma_{eff} \frac{\partial^2 \varphi}{\partial x^2}(x, t) = a_{n,p} C_{dl} \frac{\partial \eta}{\partial t}(x, t) + a_{n,p} \cdot i_0 \cdot K(c_s, c) \cdot E(c_s, \eta). \tag{50}$$

Applying the same transformation, which is used in (21)–(24) and rewriting (50) to the dimensionless form we reach

$$\frac{\partial^2 \varphi^*}{\partial X^2}(X, \tau) = \frac{\kappa_{eff}}{\kappa_{eff} + \sigma_{eff}} \left(\frac{\partial U}{\partial \tau}(X, \tau) + \nu^2 J_2(U, C_S, C) \right). \tag{51}$$

The expression in brackets on the right hand side can be calculated from the rearrangement of the canonical form (25) which yields

$$\frac{\partial^2 \varphi^*}{\partial X^2}(X, \tau) = \frac{\kappa_{eff}}{\kappa_{eff} + \sigma_{eff}} \left(\frac{\partial^2 U}{\partial X^2}(X, \tau) + \mu \frac{\partial^2 lnC}{\partial X^2}(X, \tau) \right). \tag{52}$$

This differential equation can be integrated once by X

$$\frac{\partial \varphi^*}{\partial X}(X, \tau) = \frac{\kappa_{eff}}{\kappa_{eff} + \sigma_{eff}} \cdot \left(\frac{\partial U}{\partial X}(X, \tau) + \frac{1}{C(X, \tau)} \frac{\partial C}{\partial X}(X, \tau) \right) + K(\tau) \tag{53}$$

where the integration $K(\tau)$ can be determined from the original boundaries (13), i.e., $X = 1$ and we get

$$K(\tau) = -\frac{\kappa_{eff} \delta(\tau)}{\kappa_{eff} + \sigma_{eff}}. \tag{54}$$

Integrating (53) again by X on interval [0, 1], we reach for the potential difference

$$\varphi_1^*|_{X=1} - \varphi_1^*|_{X=0} = \frac{\kappa_{eff}}{\kappa_{eff} + \sigma_{eff}} \left(U|_{X=1} - U|_{X=0} + ln \left(\frac{C|_{X=1}}{C|_{X=0}} \right) \right) - \frac{\kappa_{eff} \delta(\tau)}{\kappa_{eff} + \sigma_{eff}}. \tag{55}$$

We suppose that the dimensionless standards potential ϕ_{st} is a linear function of C_s, according to (26). Multiplying by (-1), adding $U(1, \tau)$ and $\Phi_{st}(1, \tau)$ to both sides we reach the dimensionless potential difference

$$V^*(\tau) = -\frac{\kappa_{eff} \left(U(1, \tau) - U(0, \tau) + ln \left(\frac{C|_{X=1}}{C|_{X=0}} \right) \right)}{\kappa_{eff} + \sigma_{eff}} + \frac{\delta(\tau)}{\kappa_{eff} + \sigma_{eff}} + U(1, \tau) + A + B \cdot C_s(1, \tau). \tag{56}$$

Fig. 18 shows the dimensionless half-cell potential $V^*(t)$ calculated by the reference non-splitted solution of Matlab (black dash–dot line), reverse S–M (red dashed line), sequential (green continuous line) and symmetrical (blue dotted line) splitting approximations. All of the algorithms were tested by using the parameters in Tables 3 and 4 with a macroscopic time step size of 10^{-3} and microscopic time step size of 10^{-4}.

Qualitatively all of the solutions follow the reference solution and the experimental expectations. A slight difference between the precise and the splitted solutions can be observed, especially when non-zero current is applied at the boundaries. During relaxation, the difference between all of the splitted and the reference (non-splitted) solutions decreases and becomes negligible.
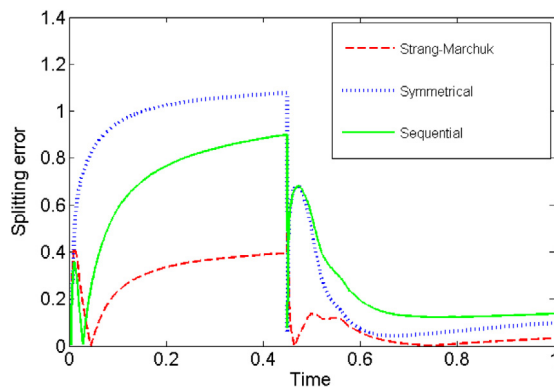
Fig. 19. Absolute error of dimensionless half-cell potential of the negative electrode.

The absolute error of the splitting schemes is shown in Fig. 19. Reverse S–M splitting (red dashed line) has the lowest absolute error, which is expected for a second order scheme according to [13]. However, symmetrical splitting (blue dotted line) is less accurate than sequential splitting (green continuous line), in spite of that fact that sequential splitting is a first order while symmetrical splitting is a second order scheme.

The absolute error in all cases levels off after a short transient. This transient may originate from the fact that when the current is switched on and off the variation of overpotential η is high, therefore the non-linear source operator varies also considerably. From an electrochemical point of view, any change in the current rapidly charges or discharges the double-layer (time dependent part in the source function (5)), which diminishes with time. Therefore consecutive current variations (i.e., during a simulation of a real driving profile of an electric vehicle) may create more inaccuracy than a constant current charge–discharge. The absolute error of the potential difference (by scaling back the dimensionless potential difference according to (18)) is ca. 25 mV, 20 mV, and 7 mV for symmetrical, sequential and S–M spitting, respectively. While 25 mV is well above the typical accuracy of the potential measurement at cell level, 7 mV is comparable to this accuracy. Consequently the reverse S–M splitting is a better choice for multi-scale models, than the commonly applied sequential splitting for modeling of LIBs.

We hypothesize that the reason why reverse Strang–Marchuk splitting is found to be most accurate is that it solves the non-linear part two times (microscopic problem) on a really fine mesh. To reach a final proof further discussions and analysis are needed.

5. Conclusion and outlook

The main aim of this paper was to investigate the applicability of operator splitting methods for multi-scale simulation of Lithium-ion batteries (LIB). First, a simplified mathematical model of a LIB's complex system was derived, which can be solved on a single or on a multiple scales. Then, the partial differential equation system of a LIB has been splitted to a linear diffusion operator and a non-linear source operator. The non-splitted reference solution was generated by Matlab's "pdepe" solver and compared with the solution of sequential, reverse sequential, symmetrical, Strang–Marchuk and reverse Strang–Marchuk splitting methods. Explicit and different IMEX numerical approximation were applied to obtain the numerical solution for the splitted operators and the full problem. The accuracy of the numerical methods was analyzed in Frobenius norm and by the comparison of half-cell potentials. The accuracy of the different splitting methods and numerical schemes was compared and it was found that

- The simulation results of the developed simplified model showed a good qualitative agreement with experimental data, therefore can be used to compare multi-scale and single scale models.
- The operator splitting methods can be applied for the multi-scale simulation of LIB because of its small numerical error and fast algorithm.
- Reverse Strang–Marchuk and symmetrical splitting methods based on IMEX numerical scheme yielded the most accurate and the fastest method, respectively.

For the sake of quantitative simulation, it is recommended to apply the original equations (4)–(16) without any simplification. The modeling domain needs to be extended to the other two parts of a LIB (i.e., cathode and separator) to reach a full scale battery model. Furthermore the particle scale source operator (microscale) can be replaced by an ordinary or partial differential system. By using operator splitting techniques a more realistic and complex system can be solved without applying computationally intensive numerical methods. It can be concluded, that the developed reverse Strang–Marchuk splitting algorithm improved the accuracy of the commonly applied vertical multi-scale models by almost a factor of three. Therefore operator splitting can be used in the next generation multi-scale models for the simulation of electrochemical devices including Li-ion batteries in a more sophisticated and precise way.

Acknowledgment

Appendix. Calculation of overpotential

Let us consider the right side of (22) and expand the bracket $\left(\frac{1}{\kappa_{eff}} + \frac{1}{\sigma_{eff}}\right)$ as follows:

$$\frac{1}{\kappa_{eff}}\left[\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\kappa_{eff}RT}{F}\cdot S(c)\right)\right] + \frac{1}{\sigma_{eff}}\left[\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\kappa_{eff}RT}{F}\cdot S(c)\right)\right].$$

Let W be chosen as:

$$W := \left[\frac{\kappa_{eff}RT}{F}\cdot S(c)\right]$$

and substituting this notation in the previous equation we get:

$$\frac{1}{\kappa_{eff}}\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{1}{\kappa_{eff}}\frac{\partial W}{\partial x} + \frac{1}{\sigma_{eff}}\left(\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{\partial W}{\partial x}\right).$$

Using the connection between conductivities in (4) and (8) we reach

$$\frac{1}{\kappa_{eff}}\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{1}{\kappa_{eff}}\frac{\partial W}{\partial x} - \frac{1}{\sigma_{eff}}\frac{\partial i_2}{\partial x}.$$

According to definitions of i_1 and i_2 developed by Newman et al. [16] we can transform the equation as follows:

$$\frac{1}{\kappa_{eff}}\frac{\partial}{\partial x}\left(\kappa_{eff}\frac{\partial\varphi_2}{\partial x}\right) - \frac{1}{\sigma_{eff}}\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial\varphi_1}{\partial x}\right) - \frac{1}{\kappa_{eff}}\frac{\partial W}{\partial x}.$$

Finally we re-substitute W and we can express φ_1 and φ_2 by η_2

$$\frac{\partial^2\varphi_2}{\partial x^2} - \frac{\partial^2\varphi_1}{\partial x^2} - \left[\frac{RT}{F}\cdot S(c)\right] = -\frac{\partial^2\eta_2}{\partial x^2} - \left[\frac{RT}{F}\cdot S(c)\right].$$

References

[1] T. Nagaura, K. Tozawa, Lithium ion rechargeable battery, Progress Batter. Sol. Cells 9 (1990) 209.
[2] A.A. Franco, Multiscale modelling and numerical simulation of rechargeable lithium ion batteries: concepts, methods and challenges, Roy. Soc. Chem. 3 (32) (2013) 13027–13058.
[3] A.A. Franco, M.L. Doublet, W.G. Bessler, Physical Multiscale Modeling and Numerical Simulation of Electrochemical Devices for Energy Conversion and Storage, Springer-Verlag, London, 2016.
[4] C. Taylor, Integrated multiscale modelling of materials, DNV GL Strategic Research & Innovation Position Paper 8, 2014, pp. 4–20.
[5] R. de Borst, Challenges in computational materials science: Multiple scales, multi-physics and evolving discontinuities, Comput. Mater. Sci. 43 (2008) 1–15.
[6] A. Havasi, J. Bartholy, I. Faragó, Splitting method and its application an air pollution modelling, Idojaras 105 (1) (2001) 39–58.
[7] M. Botchev, I. Faragó, R. Horvath, Application of operator splitting to the Maxwell equations including a source term, Appl. Numer. Math. 59 (1) (2009) 522–541.
[8] A. Kriston, G. Inzelt, I. Faragó, T. Szabo, Simulation of the transient behavior of fuel cells by using operator splitting techniques for real-time applications, Comput. Chem. Eng. 34 (3) (2010) 339–348.
[9] P. Csomos, I. Faragó, A. Havasi, Weighted sequential splitting and their analysis, Comput. Math. Appl. 50 (7) (2005) 1017–1031.
[10] J. Geiser, I. Faragó, Iterative operator-splitting methods for linear problems, Int. J. Comput. Sci. Eng. 3 (4) (2007) 255–263.
[11] I. Faragó, A. Havasi, Operator Splittings and Their Applications, Nova Science Publisher Inc., New York, 2009.
[12] I. Faragó, A. Havasi, Consistency analysis of operator splitting methods for c_0 — semigroups, Semigroup Forum 74 (1) (2007) 125–139.
[13] I. Lirkov, S. Margenov, J. Wasniewski, I. Faragó, Operator splittings and numerical methods, in: Large-Scale Scientific Computing, in: Lect. Notes Comp. Sci., vol. 3743, 2006, pp. 347–354.
[14] I. Faragó, A modified iterated operator splitting method, Appl. Math. Model. 32 (8) (2008) 1542–1551.
[15] J. Crank, The Mathematics of Diffusion, second ed., Oxford, 1975.
[16] J. Newman, K.E. Thomas-Alyea, Electrochemical Systems, third ed., Wiley-Interscience, Hoboken, New Jersey, 2004.
[17] A. Kriston, A. Pfrang, L. Boon-Brett, Development of multi-scale structure homogenization approaches based on modelled particle deposition for the simulation of electrochemical energy conversion and storage devices, Electrochimica Acta (submitted for publication).
[18] U.M. Ascher, S.J. Ruuth, B.T.R. Wetton, Implicit-explicit methods for time-dependent partial differential equations, SIAM J. Numer. Anal. 32 (3) (1995) 797–823.
[19] R.D. Skeel, M. Berzins, A method for the spatial discretization of parabolic equations in one space variable, SIAM J. Sci. Stat. Comput. 11 (1) (1990) 1–32.
[20] S.J. Harrisa, A. Timmonsa, D.R. Bakera, C. Monroeb, Direct in situ measurements of li transport in li-ion battery negative electrodes, Chem. Phys. Lett. 485 (4–6) (2010) 265–274.
[21] S. Devana, V.R. Subramanianb, R.E. White, Transient analysis of a porous electrode, J. Electrochem. Soc. 152 (5) (2005) A947–A955.