RESEARCH ARTICLE `OPEN ACCESS`

eCherry: A Modelica Library for Modular Dynamic Modelling of Electrochemical Reactors

Jan F. Pyschik[1,2] | Katharina M. Ebeling[1,2] | Georgia Ioanna Prokopou[1,2] | Michael J. Rix[1] | J. Raphael Seidenberg[1,2] | Marius B. Kleiner[1] | Matthias Leitl[1] | Alexander Mitsos[1,3,4] | Dominik Bongartz[2]

[1]Process Systems Engineering (AVT.SVT), RWTH Aachen University, Aachen, Germany | [2]Department of Chemical Engineering, KU Leuven University, Leuven, Belgium | [3]JARA-ENERGY, RWTH Aachen University, Aachen, Germany | [4]Energy Systems Engineering (ICE-1), Forschungszentrum Jülich, Jülich, Germany

Correspondence: Dominik Bongartz (dominikbongartz@alum.mit.edu)

ABSTRACT

Electrochemical reactors offer promising pathways for defossilizing the chemical industry. To understand the operation as well as to evaluate and optimise their performance and efficiency, modelling presents a crucial tool. Currently, no comprehensive open-source library for dynamic modelling of electrochemical reactors exists, and instead modelling efforts are typically specific to individual applications and lack reusability and accessibility. To address this gap, we introduce *eCherry*, an open-source Modelica library designed for fast and flexible model building of various electrochemical applications. eCherry supports dynamic modelling of multiple electrochemical reactions and experimental setups and enables integration into aggregated system models. It features modular and replaceable components, facilitating model customisation without extensive coding. Additionally, eCherry supports basic spatial discretisation using the finite volume approach and can be integrated with other open-source libraries into larger system models for enhanced functionality. This paper details the structure of eCherry, key equations, and how they combine to models, and illustrates its application via examples demonstrating its versatility.

1 | Introduction

Electrochemical reactors are gaining attention in chemical engineering since they can contribute to the defossilisation of the chemical industry [1]. To design such electrochemical reactors and surrounding processes, evaluate their performance, optimise their efficiency, and reduce the need for time-intensive and costly experimental investigations, computer-aided methods can be a valuable tool [2]. The useful outputs of computer-aided modelling include not only a better understanding of the system, but also quantitative predictions that cannot be made with experience alone. Consequently, holistic frameworks that allow the modelling of such systems are of great importance.

These frameworks should not only aim at modelling steady-state systems but also enable to model the dynamic behaviour. Many electrochemical measurement methods, such as cyclic voltammetry, are dynamic, and as a result, dynamic models are

necessary to reproduce the system behaviour. On the other hand, dynamic modelling of electrochemical reactors is also essential for studying their flexible operation, especially when connected to intermittent renewable energy sources such as wind and solar [3], as mandated for defossilisation.

Moreover, these frameworks should be structured in a modular and object-oriented way to offer the possibility of fast model generation for different electrochemical reactors. The value of such structured frameworks is already established in both the chemical engineering field [4, 5] and other fields [6].

Such an open-source dynamic framework allows the interdisciplinary study of electrochemical devices. It can be an entry point into modelling for experimental electrochemists, e.g., for parameter estimation from experimental data. Chemical engineers can use this framework to derive dynamic plant and system models to study dynamic behaviour, flexibility potential, and optimal control. Further, it can be used in grid integration and flexibility studies without requiring the electrical engineer to model the electrolyser in detail. Additionally, it can also help more experienced electrochemical modellers as a systematic modular approach reduces the time spent on model implementation. Also, the lower-level models do not have to be validated and debugged any more. Instead the modeller can focus on validating the system model.

In the literature, a large share of modelling work has been focused on specific target applications. For example, water electrolysis is a widely simulated technology due to its technical maturity. Several reviews on the modelling of proton exchange membrane (PEM) [7–11] and alkaline [11–13] water electrolysers show the current focus on deriving individual models for certain use cases and applications. Another example is the electrochemical CO_2 reduction, for which many modelling works are also available [14–17]. Nevertheless, the models reported are not easily adjustable to represent other systems. Therefore, they are less readily available for researchers wishing to build models for their specific application.

At the same time, several commercial software tools do aim to model electrochemical reactors. The commercial flowsheet simulation software Aspen Plus in its recent versions and AVEVA Process Simulation include models of water electrolysers. One of their main disadvantages is that these models are again tailored to specific water electrolyser setups (such as alkaline water electrolysis) and are not extensible. Therefore, only the implemented specific electrolyser setup can be modelled. Further, these software tools are closed-source, and expensive licenses are needed.

Besides commercial software tools, open-source toolboxes for steady-state modelling of different aspects of electrochemical processes exist in the literature. Järvinen et al. [18] published a toolbox to fit polarisation curves of water electrolysers. Folgado et al. [19] published an open-source MATLAB and Simulink toolbox to model polarisation curves for water electrolysis using different model formulations to allow their quick comparison. However, both these toolboxes only enable steady-state modelling and focus on the aspect of polarisation curves. Moreover, in the field of computational fluid dynamics, there are open-source

libraries for detailed modelling of fluid flows and transport phenomena of fuel cells and electrolysers [20], as well as libraries with a specific focus on solid oxide fuel cells [21, 22] in steady-state.

In the dynamic modelling of electrolysers, the focus is mostly paid to specific use cases. Kreitz et al. [23] modelled a dynamic power-to-gas process with alkaline water electrolysis. Webster and Bode [24] implemented a dynamic PEM electrolyser model in Modelica for integration in aggregated system models. Similarly, Kim et al. [25] implemented a dynamic model of solid oxide electrolysis in Modelica to be integrated into systems models and study control systems. There are also a few dynamic modelling libraries that are not tailored to specific use cases. Koksharov et al. [26] developed a library for modelling of electrolysers and fuel cells ranging from 0D to 3D models (steady-state and dynamic) to investigate the influence of the geometric layout on the cell performance. However, the library focuses more on detailed phenomena in the field of computational fluid dynamics. Matejak et al. [27] published the open-source Modelica library `Chemical` for modelling chemical and electrochemical processes. However, the `Chemical` library is tailored to simulate detailed electrochemical phenomena in human cells and cellular environments. As a result, it is not suitable for modelling electrochemical reactors and investigating the underlying kinetic phenomena.

In summary, the frameworks reported in the literature in the field of electrochemical processes are either tailored to specific use cases and do not allow for fast model building for various use cases, or are frequently limited to steady-state applications. Moreover, most of the modelling works focus on water electrolysis and cannot be directly transferred to different electrochemical setups. Therefore, there is a need for a more flexible library containing modular and reusable dynamic models of electrochemical reactors and systems.

To address this gap, we propose the open-source **Electroche**mical **R**eacto**r** **D**ynamics Modelica Library (**eCherry**) for fast model building for electrochemical applications. eCherry is written in Modelica. eCherry follows the spirit of our earlier work, e.g., [28], and allows the user to quickly build models of different electrochemical applications with replaceable components, building blocks, and model equations. eCherry has various use cases like modelling experimental setups, modelling dynamic behaviours of electrochemical plants, or investigating multi-reaction systems. Examples of applications of eCherry include water electrolysis or CO_2 electrolysis. Additionally, integrating electrochemical reactors into other aggregated system models is possible. The user can choose the level of detail and the models to describe the desired system without requiring extensive programming knowledge. Furthermore, the code base is open-source and accessible, allowing its extension and modification when necessary.

Electrochemical reactors consist of different physical and chemical phenomena. In the literature, such systems are called "multiphysics" [29, 30] or "multi-domain" [31, 32]. We will use the latter and introduce three domains: (i) material, (ii) electrical, and (iii) thermal domain (see Section 2.1). eCherry can be used to model the behaviour of electrochemical reactors in all three domains.

eCherry consists of dynamic models. Steady-state applications can be modelled by simulation from suitable initial conditions till a steady state is reached. All single-domain models are lumped models, i.e., 0D in spatial coordinates. Spatial discretisation can be achieved by aggregating these base models with transport layers, equivalent to the finite volume approach [33].

Thanks to its modular structure, eCherry can be optionally coupled with other open-source libraries, though some code modifications may be required. As examples, eCherry could be coupled with a library for modelling reaction and phase equilibria in transient aqueous electrolyte systems by Bremen et al. [28] and a library for modelling photovoltaic (PV) systems to calculate the design- and irradiation-dependent transient power output of a PV plant by Brkic et al. [34]. The coupling of eCherry with the PV systems library by Brkic et al. [34] will be presented in this paper.

The rest of the paper is structured as follows. In Section 2, the structure of eCherry and the implemented equations and models are presented. Examples of different applications of eCherry are given in Section 3, and conclusions and an outlook are given in Section 4.

2 | Modelling

In the following, we introduce the key modelling principles and the modelling language Modelica in Section 2.1 and provide an overview of the structure of the eCherry library in Section 2.2. The single-domain models are described in Section 2.3 and the multi-domain models in Section 2.4. We explain how the user can provide the parameter values to model certain electrochemical reactors in Section 2.5. Concluding this section, we give a short outline on creating electrochemical reactor models using eCherry in Section 2.6.

2.1 | Modelling Principles and Language

To build eCherry, we use Modelica [35, 36], an open-source, acausal, and object-oriented modelling language. Acausal modelling offers greater flexibility than conventional programming languages. The latter assign a value to each variable based on an explicit statement and require a specific execution order of the assignments. To represent equations, the use of residuals is required. In contrast, acausal modelling allows the user to describe a system with equations as one would on a piece of paper without having to define how or in which order they are calculated and solved. The engine running the simulation will then manipulate the equations symbolically to simplify and potentially manipulate the equations symbolically into a suitable form for solving them numerically. This means that the user can switch between variables provided by the user and variables calculated whilst solving the model, as long as the system is fully determined. Therefore, Modelica is well suited for the decomposition and aggregation modelling approach and enables flexible structuring on different levels. Models can be connected either by using the Modelica graphical user interface (GUI), included in most Modelica simulation environments, or by writing the connections in code.

To illustrate the modelling principles, Figure 1 shows an example of creating a model of an electrochemical reactor with eCherry using the GUI in the Modelica integrated development environment (IDE) *Dymola* [37]. The graphical representation of the reactor model consists of different blocks connected by lines. These blocks contain models of parts of the electrochemical reactor. We call these graphically visualised models *unit models* or just *units*. These unit models can be aggregated to an electrochemical reactor model using connectors (i.e., structured interfaces in Modelica through which energy or material is exchanged), visualised by lines.

This illustrative electrochemical reactor consists of unit models for the in- and outflow in the electrolyte compartments, the holdup (i.e., fluid volume, constant due to geometrical constraint) in the electrolyte compartments, the electrodes at which the electrochemical reactions take place, the diaphragm, the electrical components source and ground, and the connectors of all three domains. Simulating this example will be presented in Section 3.1.

We follow the approach of object-oriented and structured modelling [4, 38], whereby high-level systems are decomposed into a multitude of lower-level systems. Models describing these systems on the lowest level of complexity are known as irreducible wholes [39]. We describe the irreducible wholes with model equations. These irreducible wholes can then be aggregated, i.e., combined and connected, to form further lower-level models, which can again be aggregated further to form complex systems on the highest level. This decomposition and aggregation approach and reusing code by extending from low-level models correspond to the object-oriented modelling principle [4]. This principle implies that every model equation is (ideally) only written once and reused whenever possible. This reduces redundancy and simplifies adding, changing, and removing model equations.

Applied to eCherry, we decompose the electrochemical reactor (Figure 1) into models on the lowest level describing only one physical or chemical phenomenon. Electrochemical reactors as modelled in eCherry consist of three domains covering different physical and chemical phenomena: the (i) material, (ii) electrical, and (iii) thermal domain. The models on the lowest level consist of only one domain. More complex lower-level, unit, and reactor models can span one or multiple domains. These complex single- and multi-domain models are created by aggregation through inheritance from or composition of these lowest-level models. In the GUI, see Figure 1, the domains are only visible in the connectors. Unit models consist of the domains for which they have connectors.

Figure 2 visualises how this decomposition and aggregation approach is applied to the example of a unit model of a three-domain electrolyte compartment. The electrolyte compartment describes holdup with transport over the compartment boundaries in the material domain and electricity flow with resistance in the electrical domain. The thermal domain is optionally added to describe the evolution of the temperature in the reactor. This three-domain electrolyte compartment inherits from the two-domain electrolyte compartment that misses the thermal domain but is otherwise identical. The two-domain electrolyte compartment model is created from two lower-level single-domain
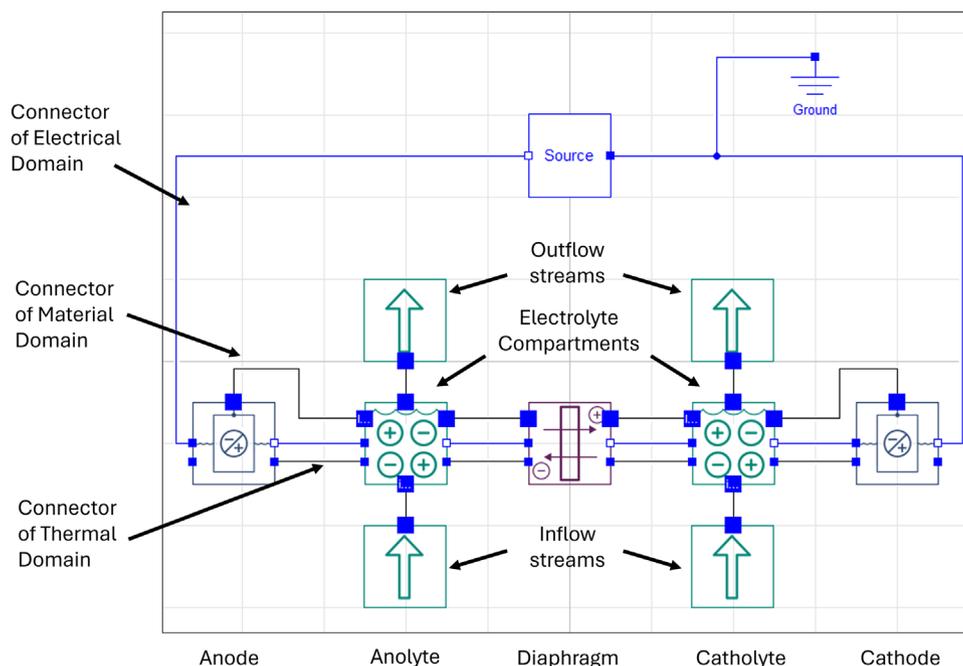
FIGURE 1 | Graphical representation of a system model of a simple electrolyser with all three physical domains (material, electrical, and thermal), created in the Modelica IDE *Dymola* [37]. IDE, Integrated development environment.
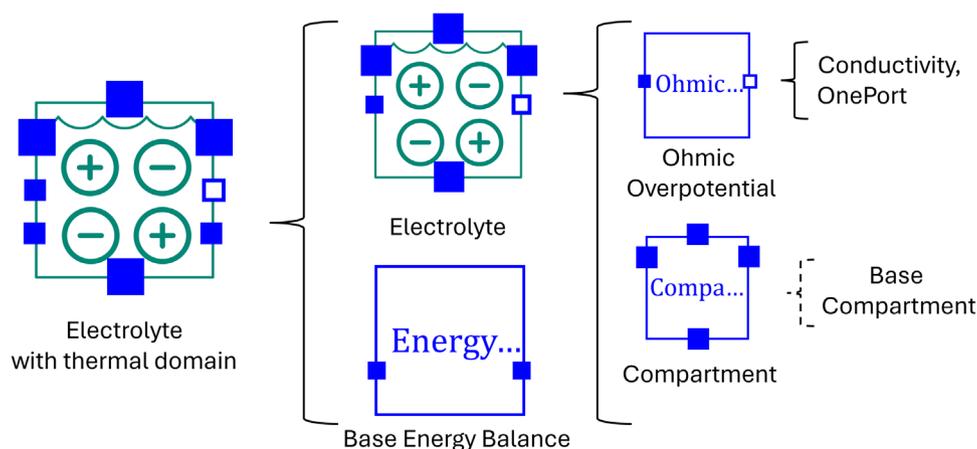

FIGURE 2 | Aggregation of an electrolyte compartment unit model by inheritance from and composition of lower-level models.

models: by inheriting a compartment model for the material domain and combining it with an ohmic overpotential model for the electric domain. Both models themselves are aggregated from further lower-level models. All models, such as for the ohmic overpotential, can also be aggregated into other high-level models. The thermal domain is added to the two-domain electrolyte by composing it with a base model of the thermal domain that provides the generalised energy conservation equation.

All models at the lowest level of complexity are lumped (i.e., zero-dimensional) models and are represented mathematically as systems of differential–algebraic equations (DAEs). Spatial discretisation is achieved by aggregating these models according to the finite-volume method [33], corresponding to discretised partial-differential–algebraic equation (PDAE(s)). The current implementation only supports 1D discretisation (1D-PDAE) but 2D or 3D is conceptually possible as well.

2.2 | Structure of eCherry

The structured modelling approach of aggregation and decomposition is also visible in the structure of eCherry. According to their complexity and functionality level, the models are sorted into different *packages* (directories). On the top level, eCherry consists of three main packages: the package `Data`, the package `Examples`, and the package `ElectrochemicalReactor`, see Figure 3.

Ready-to-simulate examples are stored in the package `Examples`. There, we provide examples of how eCherry can be used to model certain use cases. A selection of these examples is explained in Section 3.

The package `Data` provides records for input data and parameters required to define the system to be modelled. The general data records (for species, reactions, geometry, conditions, etc.) and
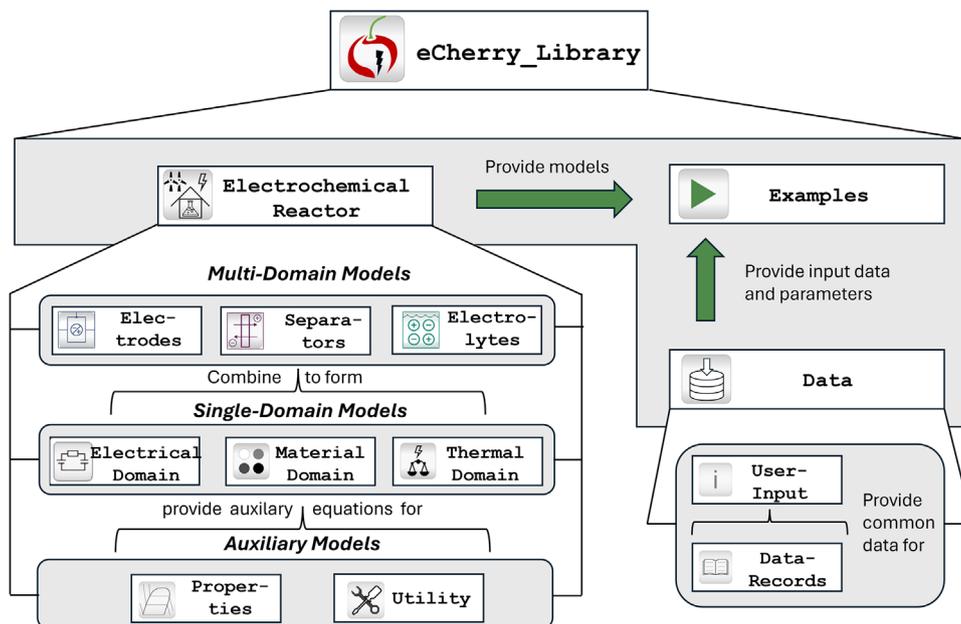
FIGURE 3 | Overview of the structure of eCherry.

default parameter values are given in `DataRecords`. To model a certain system, the user can select or manually define the required data in the package `UserInput`. More details on the different parameters and the inputs required by the user are explained in Section 2.5.

The package `ElectrochemicalReactor` contains all the unit models and their lower-level models to be aggregated to form a model of an electrochemical system (e.g., an electrolyser), compare Figure 1. Depending on the question the user wants to answer, they can either use the preexisting units or use the lower-level models to build new units.

The package `ElectrochemicalReactor` itself consists of several sub-packages that can be categorised into: (i) multi-domain models, (ii) single-domain models, and (iii) auxiliary models, see Figure 3.

The multi-domain models, see Section 2.4, combine multiple phenomena of more than one domain. Within the package `ElectrochemicalReactor`, they have the highest degree of aggregation and form key elements of electrochemical reactors. The package `Electrodes` provides models representing anodes and cathodes including their electrochemistry. Analogously, the package `Separators` collects models for membranes and diaphragms, and the package `Electrolytes` collects models for fluid phase material holdup.

One level of aggregation lower, the single-domain models, see Section 2.3, provide models that can be combined to form multi-domain models or be used on their own. For each of the three domains (i.e., electrical, material, and thermal), a certain package collects all the models that entail phenomena of that domain. The package `ElectricalDomain` contains all models that are required to model the electricity flow through and the current–voltage characteristics occurring within the electrochemical cell. The package `MaterialDomain` contains

models for all fluid phases. This includes models for in- and outflow streams, connection layers for diffusion and convection, matching connectors for material flow, and models for the electrolyte compartment within the electrochemical reactor. The package `ThermalDomain` contains models for energy conservation and heat transport. All single- and multi-domain models that incorporate material holdup or transport can be extended with these models. We provide some examples of these extensions for several multi-domain models that can be found in the respective multi-domain packages.

Since the single-domain models require more complex constitutive equations at times, we structure auxiliary models within the two packages `Properties` and `Utility`. The package `Properties` provides models for thermodynamic and kinetic properties, whilst the package `Utility` provides functions to access or calculate values to be used in the other packages.

2.3 | Single-Domain Models

In the following, we describe the single-domain models, which can either be combined to form multi-domain models or be used on their own, in more detail. All three domains consist of (i) the main model equations, i.e., integral balance equations and constitutive equations, and (ii) connecting equations that enable the handing over of variable values via connectors between different modular models.

2.3.1 | Connectors in eCherry

The units visualised in Figure 1 are connected to describe the exchange of quantities between the units, e.g., the flow of material. In Modelica, units are usually connected using *connectors*, which exchange standardised variables. Connectors only connect essential information and, thus, allow modularity and flexibility

TABLE 1 | Connectors used in eCherry (vectorised quantities are denoted by bold print).

Type	Domain	Potential variables	Flow variables
Electrical	Electrical	Electric potential (E)	Electric current (I)
Thermal	Thermal	Temperature (T)	Heat flow ($\dot{Q}$)
Material_Simple	Material	–	Molar flow rates ($\dot{\boldsymbol{n}}$)
Material_Liquid	Material	Concentrations ($\mathbf{c}$)	Molar flow rates ($\dot{\boldsymbol{n}}$)
Material_Gas	Material	Partial pressures ($\mathbf{p}$)	Molar flow rates ($\dot{\boldsymbol{n}}$)

such that any unit can be connected to any other with the same connector type. Modelica distinguishes between potential and flow variables. In the former, a difference describes a driving force, e.g., electric potential and connection sets all potential variables equal to each other. In contrast, flow variables describe the movement of a conserved quantity, e.g., electric current, and connection imposes that the sum of all flow variables is zero, following conservation of the quantity. Note that the connection ignores dynamic effects. These need to be modelled in the models being connected instead.

In eCherry, we use five connectors, which are shown in Table 1. For the electrical domain, the electrical connectors from the Modelica Standard Library are used, which connect the electric current and potential. For the thermal domain, the thermal connector `HeatPort` from the Modelica Standard Library is used that hands over a heat transfer rate $\dot{Q}$ and a temperature T. It is to be noted that any thermal resistance is not part of the connectors, but of the individual models the connectors connect. For the material domain, three different connectors are used, where the potential and the flow variable are vectors with a length equal to the number of species. Of the three, the connector `Material_Simple` only hands over the molar flow rates of all species. In contrast, the two other material connectors additionally pass an intensive quantity vector. The connector `Material_Gas` hands over the partial pressures, whilst the connector `Material_Liquid` hands over the concentrations. Concentrations are used instead of activities as these are required for the Butler–Volmer equation based on Dickinson and Hinds [40]. Under the assumption of ideal thermodynamics, they are equal to fugacities and activities. Further extensions to allow for nonideal thermodynamic calculations where required, however, are possible.

Through the structured modelling approach and utilisation of the standard connections, one does not have to write balance equations for aggregated models, since the equations generated by the connectors do that automatically. One example are the electric connectors, where the user does not need to impose Kirchhoff's circuit laws, as they are automatically satisfied. Additionally, Modelica's standard connectors allow connections to Modelica libraries for (power) electronics.

2.3.2 | Electrical Domain

The flow of current is the key distinction between electrochemical reactors and standard chemical reactors. Many parts of an electrochemical reactor are connected electrically. We adopt the passive sign convention [41], such that the power, $P = \Delta E I$ is positive if electrical energy is transformed into other types of energy (e.g., thermal and chemical) and negative if other types of energy are transformed into electrical energy.

As part of the electrical domain, we describe models for different electrical sources and the ohmic overpotential in the following. Electrodes and electrochemical reactions are discussed in Section 2.4.1, as they combine multiple domains. Each component in the electrical domain contains an equation describing their current–voltage relationship (e.g., Ohm's law, or the Butler–Volmer equation). The only exception are electric sources, as these may specify only one of the two. Lastly, as aforementioned Kirchhoff's laws are not explicitly mentioned in this section since they are adhered to as a consequence of the structured modelling approach.

Electrical Sources To be able to simulate the response of electrochemical reactors under different currents and voltages, the user can choose an electrical source. Different sources are provided as part of eCherry, but also sources from the Modelica standard library can be used. They can for example be used to specify a constant voltage or current, a linear current increase, or a cyclic voltammetry profile consisting of multiple voltage sweeps.

Ohmic Overpotential The ohmic overpotential refers to the voltage drop caused by the ohmic resistance. We express the ohmic overpotential by using Ohm's law to relate the ohmic overpotential η_{ohm} to the current I:

$$\eta_{\text{ohm}} = RI,$$

where R denotes the ohmic resistance. We use Ohm's law to calculate the ohmic overpotential of separators and electrolytes. We note that even though Ohm's law is valid for electrolytes with fairly constant concentrations, in other cases a more complex approach might be required to account for the effect of diffusion on the charge transport as covered by Newman and Thomas-Alyea [42].

For an electrolyte, the resistance can be calculated from its conductivity κ. The electric conductivity of electrolyte solutions, in general, depends on the electrolyte concentration and temperature.

Boroujeni et al. [43] investigated the accuracy of six non-electrolyte-specific conductivity models and showed that the performance decreases at high electrolyte concentrations relevant

to electrochemical applications. As a general model therefore appears impractical, we offer the option to specify a constant electrolyte conductivity or a user-defined conductivity model. As an example, we implemented an empirical correlation for the conductivity of a KOH solution as a function of temperature and molarity derived by Gilliam et al. [44]. Further conductivity models can be implemented and used accordingly. To account for the effect of void fraction (e.g., gas volume fraction) on the conductivity, we allow choosing to calculate a corrected conductivity based on the Bruggeman equation [45]:

$$\kappa_\epsilon = \kappa_0(1 - \epsilon)^{1.5},$$

in which κ_ϵ denotes the corrected conductivity, κ_0 the uncorrected conductivity and ϵ the void fraction, which is calculated as part of gas–liquid compartment models. The conductivity correlation and gas–liquid compartment models are then connected in the multi-domain electrolyte models (see Section 2.4).

2.3.3 | Material Domain

The holdup of liquid or gaseous species is modelled with lumped, fixed-volume, and ideally mixed systems, unless specified otherwise. These models are named *compartments*. A compartment unit with an ohmic overpotential model then forms an electrolyte unit which can be found in the multi-domain models below.

Material Conservation Since electrochemical reactors are mainly used for reactions, amounts of species (in moles) are used as the quantity to be balanced inside a compartment with

$$\frac{dn_{\text{tot}}}{dt} = \sum_{l=1}^{N_\Phi} \dot{n}_{\text{tot},l,\Phi} + \sum_{k=1}^{N_\Sigma} \dot{n}_{\text{tot},k,\Sigma}. \tag{1}$$

Thereby, the rate of change in the total amount of substance n_{tot}, is a result of all the flows crossing the system boundary, denoted by $\sum \dot{n}_{\text{tot},l,\Phi}$, and the amount of substance created within the compartment $\sum \dot{n}_{\text{tot},k,\Sigma}$. We explain the contributions to the two terms with constitutive equations in more detail below.

For numerical reasons, we use total mole balance Equation 1 along with $N_s - 1$ species mole balances with N_s being the number of species:

$$\frac{dn_i}{dt} = \sum_{l=1}^{N_\Phi} \dot{n}_{i,l,\Phi} + \sum_{k=1}^{N_\Sigma} \dot{n}_{i,k,\Sigma} \quad \forall\, i = 1, \dots, N_s - 1. \tag{2}$$

Within the vector quantity **n** describing the amount of substance, the solvent (commonly water) should be placed as the last species, for numerical reasons [28]. For the closure relation, all amounts of substance are summed up to the total amount of substance:

$$n_{\text{tot}} = \sum_{i=1}^{N_s} n_i.$$

Material Transported Over the System Boundaries The first term of the mole balance equations above (Equations (1) and (2)) is the sum of the flows of matter entering and exiting a compartment. Their values are determined outside the compartment models by the connected units. These can be transport equations like (i) diffusion or (ii) forced convection and (iii) electrochemical reactions.

For the diffusive transport, we use a connection layer model, which has no material holdup but describes the flow between two compartment units, and is connected to these compartments via the material connectors described in Section 2.3.1. This connection layer model, named `DiffusiveConnectionLayer`, contains Fick's law

$$\frac{\dot{n}_i}{A} = -D_i \frac{\Delta c_i}{\Delta x} \quad \forall\, i = 1, \dots, N_s,$$

where the flow of matter for one component $\dot{n}_i$ per cross-section A is a function of the diffusion coefficients D_i and the concentration difference Δc_i over a distance Δx. We note that this is a simplified treatment of mass transport that is not generally valid, as it assumes diffusion to be the main transport mechanism and neglects migration fluxes and the condition of electroneutrality as covered in Newman and Thomas-Alyea [42]. To include migration, eCherry also contains the Nernst–Planck model [42]. Whilst the Stefan–Maxwell–diffusion model is the more rigorous approach that includes thermodynamic nonidealities [46], the majority of modelling of electrochemical systems uses the simpler Fick's law [33]. Adding more models for material transport is straightforward: the discretised equations describing the diffusive flows need to be implemented into a pre-existing connection layer model.

Regardless of the transport model embedded, connection layers combined with compartments aggregate to discretised units for boundary layers.

To provide forced convective material inflows or outflows, there is a package named `Flows`. These can be used to connect to a compartment collecting all products produced or to provide (material) boundary conditions by setting a vector of fixed molar flow-rates. Adding a simple model for a pump is also conceivable.

Electrochemical reactions are also treated as transport over the system boundaries as they commonly occur at a phase boundary, namely the surface of the electrode [42], and not inside a homogeneous phase. Thus, they are modelled as occurring within the model of the electrode and not within the compartment (see Section 2.4.1). Modelling the electrochemical reactions outside the compartments allows to keep the compartment model simple and reusable for discretised modelling.

Material Produced or Consumed in the System The second term of the mole balances (Equations (1) and (2)) $\sum_i \dot{n}_{i,\Sigma}$ is a source or sink term. If the only reactions are electrochemical ones taking place at the electrodes, this term equals zero. However, homogeneous reactions within the bulk liquid phase are of special interest in applications like CO_2 reduction. The models for homogeneous reactions, unlike their electrochemical counterparts, are contained within special compartment units.

In compartments containing homogeneous reactions, both kinetic and equilibrium reactions are modelled following the

systematic approach by Walz et al. [47], where the species transformation rates R_i are calculated from the reaction rates r_r and stochiometric coefficients $\nu_{i,r}$ of a set of N_{reac} independent reactions:

$$R_i = \sum_{r=1}^{N_{\text{reac}}} \nu_{i,r} r_r \quad \forall\, i = 1, \dots, N_s \quad r = 1, \dots, N_{\text{reac}} = N_{\text{kin}} + N_{\text{equ}}$$

The species transformation rates, in turn, are proportional to the source terms in the balance equations. The reaction rates r_r are described through further constitutive equations. One option is to model them kinetically of the general form

$$r_p = k_p(T) \prod_{i=1}^{N_s} c_i^{\nu_{i,p,\text{kin}}} \quad \forall\, p = 1, \dots, N_{\text{kin}},$$

where for one kinetic reaction p, $k_p(T)$ denotes the reaction rate constant and $\nu_{i,p,\text{kin}}$ the partial orders of reaction. The other option is to model them implicitly through equilibrium relations of the general form

$$K_q = \prod_{i=1}^{N_s} c_i^{\nu_{i,q,\text{equ}}} \quad \forall\, q = 1, \dots, N_{\text{equ}},$$

where for one reaction q, $\nu_{i,q,\text{equ}}$ describes the stoichiometric coefficients and K_q the equilibrium constant of this law of mass action. Equilibrium reactions are implemented in eCherry; however, they require tailored initial conditions for the chosen case study to be solved numerically. This follows from the hidden constraints introduced by a high index DAE system, which reduces the freely selectable initial concentrations [47].

As another possibility to model the reactions in the liquid phase, we integrate our open-source modelling framework `ElectrolyteMedia` [28] into compartment models of eCherry. The `ElectrolyteMedia` framework provides an equation-oriented approach to model reaction and phase equilibria in transient aqueous electrolyte systems. The equilibria are calculated by an embedded Gibbs free energy minimisation problem via a reformulation of the Karush–Kuhn–Tucker conditions. Its advantage lies in reformulating the reaction network as a well-posed index-1 DAE. Within eCherry, the `ElectrolyteMedia` is integrated optionally into compartment units of eCherry, hence enabling exact calculation of nonideal reaction and phase equilibria together with the DAEs of a dynamic electrochemical system. For more details on the mathematical equations of the `ElectrolyteMedia` framework, we refer to Bremen et al. [28]. For the user, it is to be noted that species in each library are structured and entered differently so that the user has to specify all species twice.

Compartment Types Based on the above model equations, a number of specific compartment types are implemented in eCherry. They differ in the type of connections or the phases they contain. On one hand, the compartments are split in `Batch` and `Continuous`, alluding to the electrochemical setups batch or continuous flow cell. Both compartment types have two connectors for diffusive flows; the latter additionally has connectors for convective flows to add a forced in- and outflow. These (simple) convective flows are the one exception to the

rule of all flows crossing the system boundary being defined outside the compartment model. In this case, the flow exiting the compartment is a degree of freedom that is determined by all partial volumes within the compartment adding up to the geometrical volume, assuming isobaric conditions.

By phase, the compartments are split in either liquid-only compartments or compartments containing a gas and liquid phase. The former entails the assumption that any kind of gas is completely dissolved within the liquid phase. Additionally, liquids are assumed as ideal, incompressible liquids with a fixed density. Species dissolved in liquids are assumed as ideal dissolved species with no impact on the density.

The models with material holdup including both a gaseous and a liquid phase are assumed to be ideally mixed and isobaric. Since we designed these material holdup models to represent the fluid electrolyte compartment in an electrolyser, we modelled them with a fixed geometric volume V_{geo}. This volume constraint adds an equation required to calculate the additional variable of gaseous holdup V_G in the compartment:

$$V_{\text{geo}} = V_L + V_G.$$

The gaseous holdup V_G is approximated by the ideal gas law:

$$pV_G = n_G RT. \tag{3}$$

We have also implemented a nonisobaric compartment in which the evolving gas within the constant geometric volume leads to a pressure rise.

Since we assume ideal mixing of the gas phase, the composition of the gas leaving the compartment is identical to that of the gas phase in the compartment:

$$y_i = \frac{\dot{n}_{\text{out},G,i}}{\dot{n}_{\text{out},G}} \quad \forall\, i = 1, \dots, N_s$$

This equation, together with the material balance for each species (Equation 2), determines the individual species molar flow leaving the gaseous compartment $\dot{n}_{\text{out},G,i}$ due to the isobaric and isochoric conditions. For the compartment with continuous inflow and outflow, we assume an ideally mixed stream of both gaseous and liquid species leaving the compartment.

We do not account for pressure drop in the compartments. In most aqueous systems, the energy demand for electrolyte recirculation is negligible compared to the overall energy demand. Further, the pressure drop depends strongly on the system at hand and its flow geometries, impeding generalised model building. If the pressure drop is of interest, the user can add a tailored pressure drop block before or after the electrolyte compartments.

2.3.4 | Thermal Domain

The package `ThermalDomain` provides additional models accounting for the conservation and transport of energy in terms of heat, work, or enthalpy flow within the components of the electrolyser and in exchange with the environment. The

thermal domain is optional in the sense that the isothermal operation of the electrochemical system without calculating the heat exchange required to maintain a constant temperature in space and time is the default. By adding the thermal domain, the user can decide to additionally calculate the required cooling and heating, or model nonisothermal operation in both time and space dependency. We implemented this additional feature as another domain that can be added as an extension to the multi-domain models of the electrochemical system.

Energy Conservation Energy conservation is calculated via the general energy balance:

$$\frac{dU}{dt} = \sum_{k=1}^{N_{H,\Phi}} \dot{H}_{k,\Phi} + \sum_{j=1}^{N_{Q,\Phi}} \dot{Q}_{j,\Phi} + \sum_{l=1}^{N_{W,\Phi}} P_{l,\Phi}, \qquad (4)$$

with U denoting the inner energy of the system. A change in the inner energy of the system can either be achieved by (i) enthalpy transported due to material flow over the system boundaries, denoted by $\sum \dot{H}_{k,\Phi}$, (ii) heat transfer over the system boundaries, denoted by $\sum \dot{Q}_{j,\Phi}$, or (iii) work transfer over the system boundaries, denoted by $\sum P_{l,\Phi}$. The contributions to the three energy flow terms will be explained in the next sections.

The inner energy of the system is a function of the enthalpy H, pressure p, and volume V of the system:

$$U = H - pV.$$

We calculate the enthalpy of the system in a case-dependent manner, depending on the aggregate state of the system to be modelled. For fluid materials, the enthalpy of the system is calculated as the sum of the temperature-dependent molar species' enthalpies $h_i(T)$ times the molar amounts of species n_i:

$$H = \sum_{i=1}^{N_s} n_i h_i(T).$$

For the temperature-dependent molar species' enthalpies, we provide a calculation based on the molar species' heat capacities $c_{p,i}^j$ and the reference molar species' enthalpy $h_i(T_{\text{ref}})$:

$$h_i(T) = h_i(T_{\text{ref}}) + c_{p,i}^1(T - T_{\text{ref}}) + \frac{c_{p,i}^2}{2}(T^2 - T_{\text{ref}}^2) + \frac{c_{p,i}^3}{3}\left(T^3 - T_{\text{ref}}^3\right)$$
$$+ \frac{c_{p,i}^4}{4}\left(T^4 - T_{\text{ref}}^4\right) + \frac{c_{p,i}^5}{5}(T^5 - T_{\text{ref}}^5).$$

For pure solid materials such as electrodes and membranes, we calculate the enthalpy of the system based on density ρ, volume V, and their specific heat capacity c_p:

$$H = \rho V c_p(T - T_{\text{ref}}).$$

Enthalpy changes due to phase changes (e.g., enthalpy of vaporisation) are implicitly included via the change of species state (i.e., via the enthalpy of formation of the species in the respective states). Rigorous phase change models can be included by using the compatible ElectrolyteMedia library. At the current version of eCherry (v1.1.0), we do not offer simple phase change models apart from phase changes as part of reactions.

Enthalpy Transportation by Material Flow Over the System Boundaries For the transport of energy by material flow, we calculate a total enthalpy flow by summing over the species' molar flows and their enthalpies entering or leaving the compartment:

$$\dot{H} = \sum_{i=1}^{N_s} \dot{n}_i h_i.$$

Heat Transported Over the System Boundaries The heat transfer rate $\dot{Q}$ between interfaces of two compartments or between the interface of a compartment and the environment can be due to one or multiple of the following phenomena: (i) conduction, (ii) convection, or (iii) radiation.

For conduction, i.e., the transport of heat without the transport of material by the mechanical contact; and convection, i.e., the transport of heat by transport of material, both due to temperature gradients ΔT, the thermal conductance equals the product of an interface-dependent heat transfer coefficient α times the area A of the interface:

$$\dot{Q}_{\text{cond}} = \alpha A \Delta T.$$

The values for α are implemented as input parameters and thus need to be assigned by the user according to the surface geometry and texture, as well as the type and velocity of the fluid flow.

The convective heat losses from the surface of the housing of the electrolyser at temperature T to the environment at temperature T_{env} are implemented as follows:

$$\dot{Q}_{\text{conv}} = C_{\text{housing}}(T - T_{\text{env}}),$$

with C_{housing} denoting the thermal conductance of the housing. We calculate the thermal conductance of the housing as a function of the heat transfer coefficients, thermal conductivity, and geometry. Thereby, we provide a model that approximates the geometry of the housing as that of a pipe with the same cross-sectional area as the electrolyser. Other models can be implemented instead as well.

The effect of radiation, i.e., the transport of heat by electromagnetic waves, is in many cases negligible. Nevertheless, we implemented it for the housing of the electrolyser to estimate its effect that might become more relevant for high-temperature electrolysers:

$$\dot{Q}_{\text{radiation}} = \epsilon \sigma A(T^4 - T_{\text{env}}^4),$$

with σ denoting the Stefan–Boltzmann constant, ϵ the degree of emission, and A the surface area of the radiating system at temperature T to the environment at temperature T_{env}. We provide a model that approximates the area of the housing as that of a pipe.

In addition to these phenomena, other not-specified external heat flows entering or leaving an arbitrary component can be directly specified with a value. This way, it is also possible to calculate the effect of further not modelled phenomena on the heat-up behaviour of an electrolyser, like shunt currents, or to calculate

the heat that is required to heat or cool the system to maintain a constant temperature.

Work Transported Over the System Boundaries Work is applied to the system in the form of electric energy, for which it is defined as the product of potential difference ΔE and current I:

$$P = I\Delta E.$$

Thus, this term only applies to models that include the electrical domain. The potential difference and the current are calculated in their respective electrical domain models, which are explained in Section 2.4.1, and can be obtained from the respective electrical connectors. Whilst this approach properly calculates the total amount of generated heat in an electrochemical cell, so far it does not properly take into account the distribution of the generated heat due to electrochemical reactions to both electrodes as presented in Newman and Thomas-Alyea [42].

2.4 | Multi-Domain Models

This section includes multi-domain models, which combine multiple phenomena of more than one domain and are created by combining single-domain models.

2.4.1 | Electrodes

In the following, we explain the models to describe electrochemical reactions at electrodes. First, the models for equilibrium potential and activation overpotential are presented. Afterwards, the model for an electrochemical reaction is described, which uses both the equilibrium potential and activation overpotential as a submodel. Finally, the model of an electrode, which contains one or multiple electrochemical reactions is introduced. The required parameter values to model the electrochemical reactions are specified as part of the reaction structure described in Section 2.5.

Equilibrium Potential The equilibrium potential is calculated based on the Nernst equation. The Nernst equation can be derived from the first and the second law of thermodynamics assuming isothermal, reversible, and nonadiabatic operation. It is used to calculate the equilibrium potential ($E_{eq,r}$) of an reaction r, which is the minimum potential needed:

$$E_{eq,r} = E_{eq,ref,r} - \frac{RT}{z_r F} \sum_{i=1}^{N_{s,r}} \nu_{i,r} \ln(a_i).$$

Therein, $E_{eq,ref,r}$ is the equilibrium potential at the reference conditions, R is the ideal gas constant, T is the absolute temperature, z_r is the number of transferred electrons, F is the Faraday constant, $N_{s,r}$ are the number of species participating in reaction r, $\nu_{i,r}$ is the stoichiometric coefficient of species i in the reaction (written as reduction) and a_i is the activity of species i. The second term describes the difference in potential between the reference composition (i.e., unit activity) and the current system composition.

Unless the `ElectrolyteMedia` framework is integrated, we use ideal thermodynamics with $a_i = \frac{c_i}{c_{i,ref}}$ for dissolved species, $a_i = \frac{p_i}{p_{i,ref}}$ for gaseous species and $a_i = 1$ for solvent species. As reference conditions, we use standard conditions of $c_{i,ref} = 1$ mol/L and $p_{i,ref} = 1$ atm.

In general, $E_{eq,ref,r}$ depends on the temperature. In eCherry, it is possible to either set $E_{eq,ref,r}$ to be temperature independent, i.e.,

$$E_{eq,ref,r} = E_{eq,ref,r}(T_0) = \text{const},$$

specify an empiric formula for the temperature dependence, i.e.,

$$E_{eq,ref,r} = E_{eq,ref,r}(T),$$

or specify a constant entropy of reaction (ΔS_r), which is then used to account for the temperature dependence according to

$$E_{eq,ref,r} = E_{eq,ref,r}(T_0) + (T - T_0)\frac{\Delta S_r}{z_r F}.$$

The parameter values for the equilibrium potential at the reference conditions and temperature $E_{eq,ref,r}(T_0)$, the number of transferred electrodes z_r, the stoichiometric coefficients $\nu_{i,r}$ and the entropy of reaction ΔS_r are specified in the reaction record (see Section 2.5), where additionally the desired option for the temperature dependence of the equilibrium potential can be chosen.

Activation Overpotential The activation overpotential represents the additional required potential as the driving force for the reaction kinetics. To calculate the activation overpotential, the user can choose either the Butler–Volmer or the Tafel equation.

The Butler–Volmer equation is implemented based on the recommendation of Dickinson and Wain [48] as

$$j_r = j_{0,ref,r}\left[g_{a,r}e^{\frac{\alpha_{a,r}F}{RT}\eta'_{act,r}} - g_{c,r}e^{-\frac{\alpha_{c,r}F}{RT}\eta'_{act,r}}\right] \quad (5)$$

with $\eta'_{act,r} = E - E_{eq,ref,r}$,

where $g_{a,r}$ and $g_{c,r}$ are defined as

$$g_{a,r} = \prod_{i\in\{i|RO_{i,r}>0\}} l_i^{RO_{i,r}}$$

$$\text{and } g_{c,r} = \prod_{i\in\{i|RO_{i,r}<0\}} l_i^{-RO_{i,r}}.$$

Therein, we use $l_i = \frac{c_i}{c_{i,ref}}$ for dissolved species, $l_i = \frac{p_i}{p_{i,ref}}$ for gaseous species and $l_i = 1$ for solvent species with the same reference conditions as defined before. Further, $RO_{i,r}$ is the reaction order of species i, $j_{0,ref,r}$ is the exchange current density at reference conditions, $\alpha_{a,r}$ is the anodic transfer coefficient, $\alpha_{c,r}$ is the cathodic transfer coefficient, and $\eta'_{act,r}$ is the activation overpotential with respect to the equilibrium potential at reference conditions, for which we use the same conditions as for the Nernst equation. It should be noted that as suggested by Guidelli et al. [49, 50], we do not include the number of electrons

in the terms in Equation 5. Therefore, when considering charge transfer coefficients from the literature that include the number of electrons in the exponent, the charge transfer coefficients need to be scaled accordingly. The reaction order $RO_{i,r}$ can be equal to the stoichiometric coefficient $\nu_{i,r}$ but does not have to be. The exchange current density at reference conditions $j_{0,\mathrm{ref},r}$ is calculated as a function of temperature by an Arrhenius approach with the activation energy $E_{a,r}$ as

$$j_{0,\mathrm{ref},r}(T) = j_{0,\mathrm{ref},r}(T_0)\mathrm{e}^{-\frac{E_{a,r}}{R}\left(\frac{1}{T} - \frac{1}{T_0}\right)}.$$

The Tafel equation for an anodic current density is implemented as

$$\eta'_{\mathrm{act},r} = \frac{RT}{\alpha_{a,r}F} \ln\left(\frac{j_r}{g_{a,r}j_{0,\mathrm{ref},r}}\right)$$

and for a cathodic current density as

$$\eta'_{\mathrm{act},r} = -\frac{RT}{\alpha_{c,r}F} \ln\left(\frac{j_r}{g_{c,r}j_{0,\mathrm{ref},r}}\right).$$

Based on either the Butler–Volmer or Tafel equation, the activation overpotential with respect to the equilibrium potential at the electrode surface instead of the reference conditions is then obtained as

$$\begin{aligned} \eta_{\mathrm{act},r} &= E - E_{\mathrm{eq},r} \\ &= \eta'_{\mathrm{act},r} + \frac{RT}{z_r F}\sum_{i=1}^{N_{s,r}} \nu_{i,r}\ln(a_i). \end{aligned}$$

For a discussion of the meaning of $\eta_{\mathrm{act},r}$, we refer the reader to Seidenberg et al. [51].

The parameter values for the exchange current density at reference conditions $j_{0,\mathrm{ref},r}(T_0)$, the charge transfer coefficients $\alpha_{a,r}$ and $\alpha_{c,r}$, the stoichiometric coefficients $\nu_{i,r}$, the reaction orders $RO_{i,r}$, and the activation energy $E_{a,r}$ are specified in the reaction record (see Section 2.5).

Electrochemical Reaction The electrochemical reaction model describes an electrochemical reaction at the electrode. It includes submodels for the equilibrium potential (Nernst equation) and the activation overpotential as mentioned above. The electrochemical reaction model also calculates the molar flow rates as a result of the reaction based on Faraday's law:

$$\dot{n}_{i,r} = \frac{\nu_{i,r} j_r A}{z_r F}, \qquad (6)$$

where $\dot{n}_{i,r}$ is the molar flow rate of species i as a result of reaction r, $\nu_{i,r}$ the stoichiometric coefficient of species i in reaction r, A is the area, z_r is the number of transferred electrons in the reaction r, and F is the Faraday constant.

Electrode The electrode unit includes the models of the electrochemical reactions. Here, one or multiple electrochemical reactions can be specified. For each specified reaction, an electrochemical reaction submodel is created. Simultaneous electrochemical reactions at the electrode surface are modelled such

that the total current density $j_{\mathrm{electrode}}$ is equal to the sum of the current densities of each reaction taking place at the electrode j_r, as follows:

$$j_{\mathrm{electrode}} = \sum_{r=1}^{N_r} j_r,$$

where r denotes the index of the reaction at the electrode, and N_r is the number of reactions. The total potential drop (i.e., the activation overpotential and equilibrium potential) of every electrochemical reaction is set equal to the electrode potential drop. The electrode has two electric connectors and one material connector. The electrode does not include any material holdup; the material flows resulting from the reaction are directly passed through the material connector.

For the electrochemical kinetics, we follow the International Union of Pure and Applied Chemistry (IUPAC) sign convention for current and potential, in which anodic current and potentials are defined as positive and cathodic current and potential as negative [52], which fulfills the passive sign convention mentioned in Section 2.3.2. However, for straightforward modelling, we desire to only connect positive to negative connectors within the cell on the top level. Thus, the current at both electrodes is required to have the same sign as it needs to flow in the same direction. Additionally, it has to be taken into account that all reactions are specified as reduction although some of them are running as oxidation, which requires reversing the sign of the equilibrium potential. Therefore, we reverse the signs of the current as well as the potential drop due to activation overpotential and equilibrium potential in the required cases.

2.4.2 | Electrolytes

Unit models for the electrolyte compartments are provided in the `Electrolytes` package. These models describe the holdup of the electrolyte solution in the electrolyser and the ohmic overpotential across the electrolyte. The electrolyte holdup is described by inheriting from the respective models from the material domain (Section 2.3.3). The ohmic overpotential is described by inheriting the ohmic overpotential models (Section 2.3.2). Optionally, the heat-up behaviour can be described by using `Thermal` models. Figure 2 visualises the inheritance structure of the electrolyte compartments.

The conductivity of the electrolyte solution is required to calculate the ohmic overpotential. The conductivity can either be set to a constant value or calculated with electrolyte-specific correlations.

2.4.3 | Separators

Within the `Separators` package, we provide models representing membranes and diaphragms. Separator models incorporate an ohmic overpotential from the electrical domain, as described in Section 2.3.2, and a material flow of specific species.

Currently, we provide two types of separator models: (i) a PEM and (ii) a diaphragm. Both have a specific ohmic resistance R_{memb}

respectively R_{diaph} that is calculated based on the conductivity of the separator. The conductivity σ_{sep} of the separator can either be set to a user-specified value or calculated using simple conductivity models.

For the PEM type, the conductivity is calculated based on the water content of the membrane λ [53]:

$$\sigma_{\text{sep}} = \sigma_{\text{303K}} \cdot \exp\left[1268\,\text{K} \cdot \left(\frac{1}{303\,\text{K}} - \frac{1}{T}\right)\right],$$

where σ_{303K} is the conductivity of the membrane at a temperature $T = 303\,\text{K}$, calculated via equation:

$$\sigma_{\text{303K}} = 0.005138\,\Omega^{-1}\text{cm}^{-1} \cdot \lambda - 0.00326\,\Omega^{-1}\text{cm}^{-1}.$$

The mean water content of the membrane at 303 K is a function of the water vapour activity at the membrane boundaries. It is calculated using the equation of Springer et al. [53], assuming that the membrane is saturated with water at its boundaries.

The diaphragm is a porous medium filled with the electrolyte solution. Based on Brauns et al. [54], we implemented a simple model for calculating the diaphragm conductivity σ_{sep}. The diaphragm conductivity is calculated as

$$\sigma_{\text{sep}} = \sigma_{\text{El}} \cdot \frac{\varepsilon}{\tau},$$

depending on the conductivity of the electrolyte filling the pores σ_{El} and the diaphragm porosity ε and tortuosity τ, assuming an ideal wetting of the whole pore system. The porosity and tortuosity are material constants of the diaphragm. The conductivity of the electrolyte filling the pores is assumed to be the mean of the anolyte and catholyte.

Mass transfer through the separator occurs for both separator types. We implemented two models to calculate mass transfer through the separator. In one model, we assume that the molar flow rate through the separator $\dot{n}_i$ consists of only one type of ions i and is induced only by the current flow I:

$$I = \dot{n}_i F, \quad i \in \{\text{OH}^-, \text{H}^+\}.$$

For the membrane, the type of ions equals protons (H^+), whilst for the diaphragm, the type of ions equals hydroxide ions (OH^-). In the second model, the mass transfer of each species is calculated assuming Fick's law of diffusion with a user-defined effective diffusion coefficient for each species. Thus, users can model mass transfer of individual species tailored to their application.

2.5 | Providing Data

When setting up a model for a specific electrochemical setup, the user needs to define some parameters or specifications (e.g., participating species and occurring reactions) of the system. To guide the user, these are stored in the package Data, see Section 2.2. Within the package Data, the package UserInput contains combinations of all records and parameters, which are used in the illustrative examples, see Section 3. These provide examples for the user arranging parameters for a new systems

model. The package DataRecords provides further packages and *records* (a Modelica class to structure parameters) according to the application case of the parameters required: geometry, operating conditions, thermal behaviour, species, electrochemical reactions and bulk reactions. Generally, Modelica requires that all parameters must be specified with a value at simulation time, even if it is not used. That is why we set default values for parameters that are not always required to a value causing failure of simulation if used, usually zero.

The dimensions of the parts of the electrolyser are specified in the record Geometry. Temperature and pressure (or their initial values if not constant) at which the reactor operates are specified in the record Conditions. For now, this is limited to rectangular shapes and Cartesian coordinates. If the user wants to include an energy balance, the record Thermal specifies the respective input parameters required. The remaining major packages within the package DataRecords as well as the package UserInput are explained in the following in more detail.

Species The package Species contains records of species (i.e., molecules and ions) and values for their properties. A subset of these species can then be chosen to participate in electrochemical reactions that are defined in a record called ElecReaction that will be explained below. We differentiate between four types of species: (i) GaseousSpecies, (ii) DissolvedSpecies, (iii) LiquidSpecies, and (iv) SolidSpecies. It should be noted that a molecule present in both dissolved and gaseous states is specified as two different species, e.g., dissolved hydrogen is a different species than gaseous hydrogen. Regardless of its type, each species has the same property parameters, e.g., molar mass or charge. When specifying a new species, all parameter values must be given; this is a constraint set by Modelica.

Electrochemical Reaction The package ElecReaction contains an identically named record that lists the parameters required to define electrochemical half-cell reactions. To define a half-cell reaction, the partaking species, their names, stoichiometry, thermodynamic parameters, and kinetic parameters must be specified. Each new half-cell reaction, determined by these parameters, has to be added to a List_Of_Reactions; necessitated by the generalisable assumption of having multiple reactions at each half-cell. Some common reactions with pre-filled parameter values are also provided. The usefulness of this record ElecReaction is that users only have to choose or provide the reaction data but not write the model equations themselves, as these are constructed within the models for the electrode.

Bulkphase Reaction The BulkReaction contains the data for the chemical reactions within the bulk phase. It has the same structure as the record ElecReaction, but with the difference that now either a kinetic constant k or an equilibrium constant K has to be provided, as explained in Section 2.3.3.

User Input The UserInput is the single place where the user of eCherry needs to specify the parameter values of the system. Thus, it contains instances of the just mentioned records. For most models, this includes the specification of the Species record by listing the species present in the system, their initial concentrations in the electrolyte and partial pressures, if a gas phase is present, the definition of the set of reactions from

the `List_Of_Reactions`, the definition of the `Geometry`, and the operating `Conditions`. Whilst the records `Species` and `ElecReaction` are required for any model of an electrochemical system in eCherry, using the other records is a strong suggestion as they bundle parameters, but are not strictly necessary, as they can also be set for each unit individually.

Depending on the unit models, further input parameters are required, e.g., the molar flow rate of the species in the inflow in case of a continuous flow reactor. We suggest specifying them in the `UserInput` as well, especially if they are used at multiple occurrences in the model. Even if these inputs are assigned by default to all unit models requiring them, they can be locally overwritten, i.e., a different value for a certain parameter can be set for an individual unit model. The easiest way to do this is to adjust the values in the GUI.

2.6 | Building a Model

In this section, we give a short outline of the steps involved in building a system model using eCherry. In general, the steps can be either performed in text form using the Modelica syntax or by using the GUI of a respective Modelica IDE (e.g., Dymola or OpenModelica). A practical example of the involved steps is presented as part of the first illustrative example, the alkaline water electrolyser (Section 3.1).

The building of a system model can be divided into four steps:

1. Selection of modelling units,
2. Connection of modelling units via connectors,
3. Specification of data records and input values,
4. Assignment of data records and input values.

We will cover these steps in the following.

Selection of Modelling Units First, the modelling units have to be selected. For this purpose, the physical elements of the system to be modelled have to be mapped to the corresponding modelling units. Afterwards, an instance for each modelling unit has to be created, which can either be done in text form using the Modelica syntax or by drag and drop using the GUI.

Connection of Modelling Units via Connectors After the modelling units are instantiated, they have to be connected via the connectors mentioned in Table 1. For the electric connectors, a positive connector always has to be connected to a negative connector. The connections can again be made either using the Modelica syntax or the GUI. The `Ground` model always has to be used to define the potential of $0\,\mathrm{V}$.

Specification of Data Records and Input Values As a next step, the data records and input values have to be specified as described in terms of the user input before.

Assignment of Data Records and Input Values Afterwards, the data records and input values have to be assigned to each modelling unit, which again can be done either in text form

using the Modelica syntax or using the GUI. Here, choices for submodels can also be made (e.g., to select the Tafel equation for modelling the reaction kinetics at the specific electrode). The parameter values specified as part of the assigned records provide the information for the modelling units they are assigned to. They can also manually be overwritten or specified (e.g., if the cathode geometry is different to the anode geometry).

3 | Illustrative Examples

In the following, we provide some examples to showcase how eCherry can be used to model electrochemical reactors. For the first example, we also give a detailed description of the steps involved in building a model as outlined in Section 2.6. For the remaining examples, the model setup is explained in less detail to save space and focus on highlighting other possible applications.

In the first example, we start by modelling a water electrolyser and showing how a distribution of overpotentials can be obtained (Section 3.1). Second, we show how multiple reactions occurring at the same electrode can be modelled and analysed (Section 3.2). Third, we demonstrate the dynamic capabilities of eCherry. For this purpose, we present examples investigating the temperature dynamics of an electrolyser stack (Section 3.3) and modelling cyclic voltammetry experiments as a common electrochemical measurement method for electrode testing (Section 3.4). Finally, we demonstrate how eCherry can be linked to other Modelica libraries and how the coupling of electrolysis systems with renewable energy plants can be investigated (Section 3.5). These examples are intended to show (i) what can be done in eCherry, (ii) what are limitations and (iii) what can be customised.

The models of these examples are part of eCherry and can be found in the package `Examples`.

3.1 | Alkaline Water Electrolyser

In the following, we present an alkaline water electrolyser model based on the work of Hammoudi et al. [55] and Henao et al. [56]. Part of their work is the evaluation of overpotentials for the *Stuart* electrolyser, which we reproduced using eCherry.

The used electrolyser model is the same as shown in Figure 1; however, for each compartment, the version without energy balance is used. In the following, we will go through the steps of building the model.

Selection of Modelling Units We decided on the modelling units based on the layout of the cells in Hammoudi et al. [55]. The modelling units that were selected and instantiated are two `Electrode` models representing anode and cathode, two `Electrolyte_Conti_0D_GL` (0D gas–liquid electrolyte compartments with continuous in and outflows) models representing anolyte and catholyte, a `DiaphragmHydroxide` (separator only allowing OH^- passing) model representing the diaphragm, two `Material_Simple_InFlow_Fixed` (inflows with fixed flowrates) models representing the anolyte and catholyte inflow, two `Material_Simple_OutFlow` (outflows without specifications) models representing the anode and catholyte outflow,

a `CurrentDensity_Linear` (linearly increasing current density) model as electricity source and a `Ground` model.

Connection of Modelling Units via Connectors The connections made for the case of the alkaline water electrolyser are shown in Figure 1 (when neglecting the shown connections for the energy balance). We first placed the units according to their physical connections and the direction of current flow. Afterwards, connecting the units mainly involves the connection of adjacent units.

Specification of Data Records and Input Values For the alkaline water electrolysis example, we specify a `Species` record containing the occurring species O_2, H_2, OH^-, H^+ and H_2O, where for simplicity we do not model K^+. We specify two `Reaction` records that represent the oxygen evolution reaction (OER) and the hydrogen evolution reaction (HER), for which we define the exchange current density at standard conditions, the charge transfer coefficients, the participating species, the stoichiometry, and the reaction orders. As no concentration dependence is considered in Hammoudi et al. [55] and Henao et al. [56], the reaction orders, which represent the concentration dependence, are set to zero. We evaluated the exchange current densities and transfer coefficients at the considered temperature of 322.15 K based on the temperature-dependent expressions given in their paper. As we already manually calculated the exchange current densities at the considered temperature and therefore no further temperature correction is necessary, the activation energy of both reactions is set to zero. We further specify a `Geometry` record containing the general dimensions of the electrochemical cell, a `Condition` record containing pressure and temperature, and a vector containing the initial and inlet concentrations. The key parameter values are presented in Table A3.

Assignment of Data Records and Input Values We assign the records and inputs defined before to the modelling units. We specify at the electrode representing the cathode that it is the cathode in the electrolysis mode to determine the signs of current and voltage on the top level correctly (see also Section 2.4.1). Additionally, we select the Tafel equation as activation overpotential model at both electrodes (as in Hammoudi et al. [55] and Henao et al. [56]), the empiric conductivity model for KOH electrolyte from Gilliam et al. [44], and the Bruggeman equation [45] for the influence of gas bubbles on the conductivity at both electrolytes. Finally, we specify the remaining required parameter values that were not defined in the `UserInput` before, e.g., the rate of the current increase of the electrical source used in Section 3.1. We use a very slowly increasing current density ($1\,A/m^2/s$) so that we are near the steady-state at each point in time and can obtain the polarisation curve with a single simulation, allowing us to compare to the steady-state simulations of Hammoudi et al. [55].

Calculation of bubble-related ohmic overpotential Hammoudi et al. [55] and Henao et al. [56] differentiated between the ohmic overpotential that would be caused without void fraction due to bubbles and the additional overpotentials caused by the void fraction due to bubbles. As these bubble-related overpotentials are not explicitly defined in eCherry by default, we calculate them based on the simulation results. We calculate the
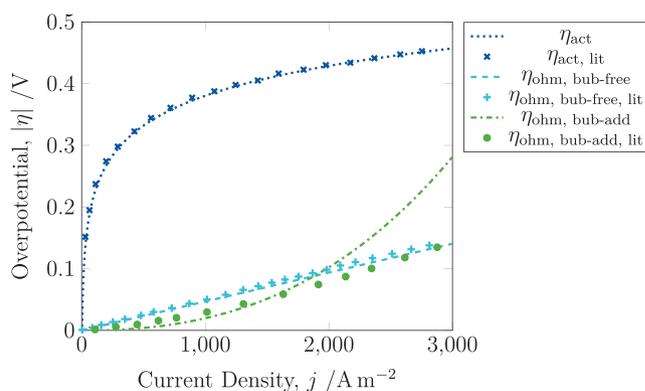

FIGURE 4 | Activation overpotential, bubble-free ohmic overpotential, and additional ohmic overpotential due to bubbles calculated based on simple alkaline water electrolyser model for the Stuart electrolyser modelled in Hammoudi et al. [55] and Henao et al. [56]. The subscript "lit" is used to refer to the simulation results from Hammoudi et al. [55].

bubble-free ohmic overpotential as

$$\eta_{\text{ohm, bub-free}} = \frac{\kappa_\varepsilon}{\kappa_0} \left(\eta_{\text{ohm, anolyte}} + \eta_{\text{ohm, catholyte}} \right) + \eta_{\text{ohm, diaphragm}},$$

and the bubble-additional ohmic overpotential as

$$\eta_{\text{ohm, bub-add}} = \left(1 - \frac{\kappa_\varepsilon}{\kappa_0} \right) \left(\eta_{\text{ohm, anolyte}} + \eta_{\text{ohm, catholyte}} \right),$$

whereas before κ_0 denotes the conductivity of the void-free electrolyte and κ_ε denotes the conductivity when taking the void fraction into account. We neglect the overpotential due to reduced active surface area due to bubbles (i.e., surface coverage).

Comparison of results The obtained overpotentials as a function of current density are presented in Figure 4. The results are similar to the results reported by Hammoudi et al. [55] for the Stuart electrolyser. The bubble-free ohmic overpotential is slightly lower than what Hammoudi et al. [55] report, which can be attributed to the fact that we neglect the ohmic potential drop within both electrodes. If desired, this could be additionally considered by adding a simple resistor model from the modelica standard library. The bubble-additional ohmic overpotential differs and is much higher at large current densities than in their work. This can be attributed to the higher void fractions predicted by our model assuming all gas is evolved in the gaseous form and transported out with the same speed as the liquid (void fraction of ca. 0.43 for anolyte and 0.6 for catholyte at 3000 $A\,m^{-2}$) compared to the void fraction value obtained by the empirical correlations used by Hammoudi et al. [55] (void fraction of ca. 0.22 at 3000 $A\,m^{-2}$ for either electrolyte). If desired, the use of such empirical correlations is also easily possible by customising the respective models: in the shown case, the conductivity model can be easily altered to use the value for the void fraction given by the empiric correlation instead.

3.2 | Multiple Electrochemical Reactions

Next, we discuss an example that can serve as a guideline on how multiple parallel reactions can be modelled within eCherry. For
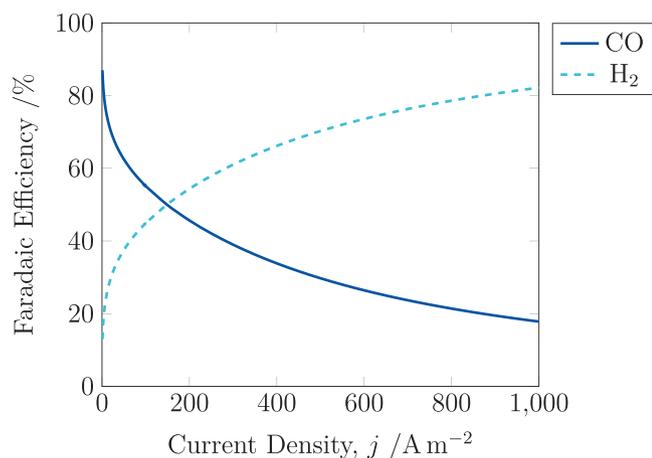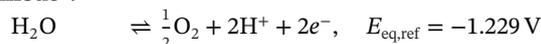
FIGURE 5 | Faradaic efficiencies of the CO_2 electrolyser of the example for modelling multiple electrochemical reactions.
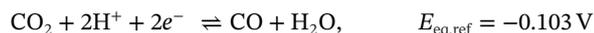
this, an electrochemical system for CO_2 reduction in an aqueous electrolyte, with two parallel reactions at the cathode, is modelled based on the work of Brée et al. [57].

The cell consists of an anode, an anodic compartment with a liquid electrolyte, a membrane, a cathodic compartment with a liquid electrolyte, and a cathode. The OER takes place at the anode assuming a Pt-electrode [57], and the carbon dioxide reduction reaction (CO_2RR) and the HER take place at the cathode, assuming an Ag-electrode [57]

Anode :

$$H_2O \rightleftharpoons \tfrac{1}{2}O_2 + 2H^+ + 2e^-, \quad E_{eq,ref} = -1.229\,V$$

Cathode :

$$2H^+ + 2e^- \rightleftharpoons H_2, \qquad\qquad E_{eq,ref} = 0\,V$$

$$CO_2 + 2H^+ + 2e^- \rightleftharpoons CO + H_2O, \qquad E_{eq,ref} = -0.103\,V$$

The set-up of the model is similar to the example in Section 3.1: for each compartment, the version without the thermal domain is used, an anode and cathode unit are used to describe the reactions taking place at the electrodes, 0D liquid compartments with continuous inlet and outlet convective flows and fixed constant conductivity are used as well as a membrane unit instead of a diaphragm. Moreover, for the description of the kinetics, the Butler–Volmer equation is chosen. Mass transport phenomena are neglected. The main difference compared to the example in Section 3.1 is that the record `ElecReaction` at the cathode contains two entries, representing the two competing reactions taking place at the cathode. The used parameters can be seen in Table A4.

Similar to Section 3.1, the overpotential distribution can be evaluated. However, the Faradaic efficiency (FE) is additionally relevant in this example, as multiple reactions take place at the cathode. Figure 5 shows the FE of the two competing reactions. The FE is computed by dividing the current density of each reaction by the total current density. At low current densities, the CO_2RR is favoured as can be seen by the high current efficiency. With an increase in the current density, though, there is a shift towards the HER, which causes an increase of FE of

HER and a decrease of FE of CO_2RR. Similar qualitative trends are observed also in Brée et al. [57]. However, the decrease of the FE of CO_2RR in Brée et al. [57] is more steep compared to the results presented here, which can be attributed to the fact that mass transport phenomena are neglected, as mentioned before. Moreover, in the work of Brée et al. [57], the Butler–Volmer equation with no concentration dependence is used, which can cause this quantitative deviation. The present model is simplified and can be extended to include mass transport phenomena in the electrolyte by discretising over the electrolyte compartment.

3.3 | Electrolyser Cold-Start

This section presents an example of the thermal domain of eCherry. For this, we demonstrate how the electrolyser model from Sakas et al. [58] can be modelled in eCherry to show the dynamic behaviour of an industrial electrolyser in terms of temperature change during a cold-start.

The model of Sakas et al. [58] represents an industrial 3 MW alkaline water electrolysis plant operating at 16 bar. They validated their results with measurements of the real plant. Their electrolyser model consists of two stacks in parallel with a bipolar configuration and a total of 326 cells. To account for mass and energy conservation, 0D balance equations are solved in their work. For temperature control in the stack, the lye is recirculated and flows through heat exchangers. In this way, the in-flowing lye temperature equals the out-flowing lye temperature during start-up or is controlled to a maximum of 70 °C. Through pumps and additional feed water, a constant inlet mass flow is ensured.

Since the two parallel stacks in the model of Sakas et al. [58] are identical with 163 cells each, we only model one of these stacks to reduce calculation effort. Each of these cells consists of electrolytes, electrodes, and a membrane. The bipolar plates are modelled by adding heat transfer between the electrodes of neighbouring cells. Heat transfer coefficients between all components were set to a high value of $100\,kW/m^2\,K^{-1}$ to approximate a spatially equal temperature distribution within the stack. The lye inflow is equally distributed to all the cells and its temperature equals the cell temperature but is restricted to a maximum of 70°C using an additional constraint. This recreates the temperature-controlled recirculation as in Sakas et al. [58]. Other model parameters, like the geometry of the stack or the material properties, are set as in Sakas et al. [58]. Also, the equations for overpotentials and heat loss are adopted according to their work.

First of all, Sakas et al. [58] investigated the steady-state temperature of the cell stack at 100 % $j_{cell,max}$, meaning full power (1717 A/m²). At first, they reached a steady-state temperature of about 76°C which did not correspond to the data from the real electrolyser. Thus, they added an additional heat flow due to shunt currents $\dot{Q}_{shunt}$ of 308.84 kW to reach the steady-state temperature of about 80 °C. They also performed a sensitivity analysis on how fast the electrolyser heats up to the maximum inlet temperature of 70 °C for 100 %, 80 %, 60 % and 40 % of $j_{cell,max}$.

Figure 6 shows the temperature of the electrolyser during the startup until it reaches steady-state temperature as simulated
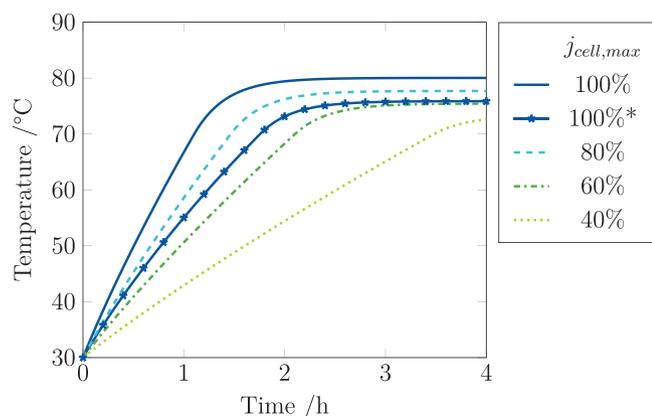
FIGURE 6 | Dynamic temperature behaviour of an electrolyser cold-start for different percentages of supply current, based on Sakas et al. [58],*:$\dot{Q}_{shunt} = 0$.

using eCherry. It can be observed that for 100% $j_{cell,max}$ almost the same steady-state temperatures are achieved for the case with shunt currents and without. For the sensitivity analysis using eCherry, the shunt currents were adapted to the same percentages as the supply current j_{cell}. The results for the heat-up time show qualitatively the same behaviour and compared to Sakas et al. [58], the heat-up times differ between approximately 10 to 18 min.

The deviations can occur due to different values used for the heat capacity of the components as we use exact values for hydrogen, oxygen, and water whereas Sakas et al. [58] only used one value for the whole electrolyte. Also, we adapted the shunt currents to the same percentages as the supply current j_{cell}, whilst Sakas et al. [58] did not mention how they adapted them. The results show that eCherry can be used to account for the dynamic and steady-state thermal behaviour of electrolysers. Furthermore, this example shows that the `ThermalDomain` adds another time constant to the system as the effects of heat transfer are observed over hours instead of seconds. Therefore, it can be useful to exclude the `ThermalDomain` if it is not necessary as it can lead to stiff differential equations.

3.4 | Cyclic Voltammetry

In this example, we show how eCherry can be used to model one of the most frequently applied dynamic electrochemical measurement methods, cyclic voltammetry. We compare to experimental cyclic voltammetry data reported by Elgrishi et al. [59] for Ferrocene/Ferrocenium (FC/FC$^+$) on a glassy carbon electrode (in their Supporting information).

As the cyclic voltammetry is conducted in a setup with a reference electrode and therefore allows to exclude the contribution of the counter electrode, we only model the half-cell including the working electrode. The main dynamics of cyclic voltammetry are in many cases determined by the changing concentrations at the surface due to limited mass transfer. We divide the electrolyte into a thin boundary layer in which a concentration gradient can be present and an ideally mixed bulk solution. To model the boundary layer, we use a discretised electrolyte unit (connection

of 500 very thin 0D electrolyte models), in which the mass transfer due to diffusion between adjacent elements is modelled by connecting them with diffusive connection layer models. This discretised electrolyte unit is connected to the electrode and a larger 0D electrolyte unit, which represents the bulk solution. A diffusive connection layer unit is used to calculate the final diffusive flows between the boundary layer and the bulk electrolyte. Finally, to complete the modelled half-cell, a cyclic voltammetry source unit is connected to the electrode and the 0D electrolyte unit and a ground model is used to set the reference value of 0 for the electric potential.

The used parameter values are collected in Table A6. We assume the reaction orders to be equal to the stoichiometric coefficients of 1 and -1, respectively. We neglect the ohmic overpotential of the electrolyte by setting a very high conductivity value ($30000\,\mathrm{Sm^{-1}}$), as the experimental data are also corrected with regard to the ohmic potential drop. For simplicity, we do not consider that in general the presence of an ohmic overpotential influences the effective scanrate that the electrode experiences. In general, an additional ohmic resistance or a conductivity model for the used electrolyte can be used to account for this.

As a reference point for choosing the total thickness of the boundary, we consider the case of constant current, for which the boundary layer thickness δ_n can be estimated as [60]

$$\delta_n = (\pi D t)^{\frac{1}{2}},$$

where D is the diffusion coefficient and t is the time. Based on that, we choose the thickness x of the part of our boundary layer model as

$$x = 10\delta_n,$$

$$\text{with } t = \tau = \frac{2\Delta E}{\nu},$$

where we use the value of $D_{\mathrm{Fc^+}}$ for D. Therein, τ denotes the time per cycle of cyclic voltammetry, ΔE denotes the total potential difference applied, and ν denotes the scanrate. The thickness of the real boundary layer (the region in which concentration changes are actually present) can be smaller than the thickness of the discretised electrolyte model representing the boundary layer and is a result of the simulation. For numerical stability, we set the area of the cell to $10\,\mathrm{m^2}$. As in the model the current scales linearly with the area (no inhomogeneities along the area are considered), we afterwards calculate the corresponding current at the area of the experiment by scaling accordingly. We use kinetic parameters given by Pournaghi-Azar and Ojani [61], which were obtained in a different study for a glassy carbon electrode in a different electrolyte.

The results of the model are presented in Figure 7 in terms of potential-current curves. Figure 8 shows corresponding concentration profiles for FC$^+$ at the three different time points marked in Figure 7. The concentration at the surface changes significantly, which ultimately gives rise to the typical cyclic voltammetry shape of the potential–current curves observed in Figure 7. Whilst the obtained potential–current curves differ quantitatively from the results reported by Elgrishi et al. [59], they
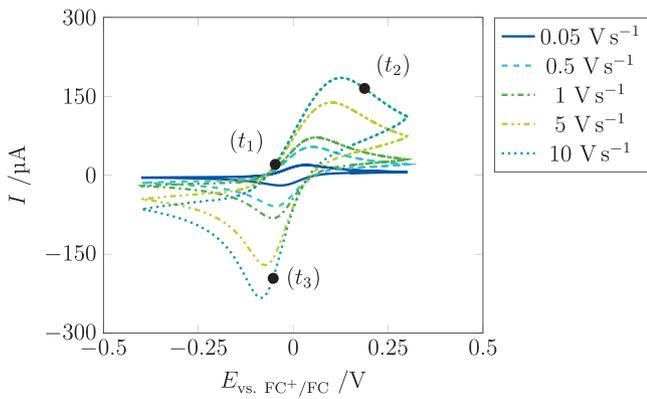
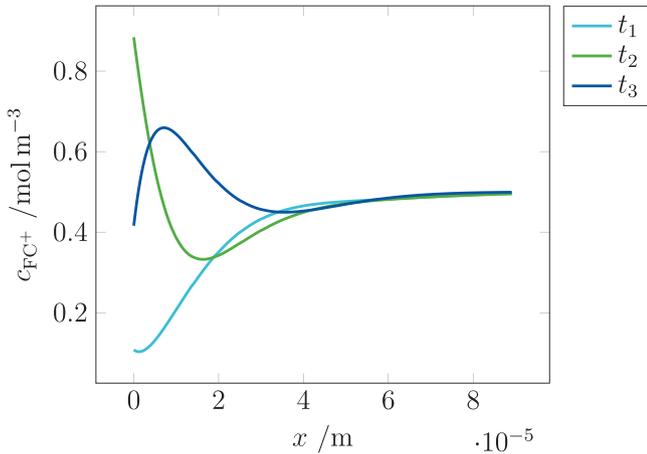**FIGURE 7** | Cyclic voltammetry at different scan rates.

FIGURE 8 | Concentration profiles during cyclic voltammetry at the different time points marked in Figure 7 for a scan rate of $10\,\text{Vs}^{-1}$. Only the first $9 \cdot 10^{-5}$ m of the discretised electrolyte layer with a total thickness of $x = 3.6 \cdot 10^{-4}$ m is shown.

look very similar qualitatively. The quantitative differences could be due to a different electrolyte and potentially varying electrode properties in the experiments of Elgrishi et al. [59] (which we compare to) and Pournaghi-Azar and Ojani [61] (where we took the kinetic parameters of) as well as the neglection of migration fluxes in the used model.

3.5 | Coupling Electrolysis and Photovoltaics

Electrolysers are expected to become an integral part of energy systems with high percentages of renewable energy. Due to the fast dynamics of electrolysers, they are well suited to be used intermittently [62] and to provide ancillary services for grid stabilisation [63, 64]. Whilst eCherry can provide the dynamic model for the electrolyser, the standard Modelica electrical connectors allow connections to a variety of power electronics models for which many Modelica libraries already exist [34, 65].

As one example, the open-source *PhotoVoltaics* library by Brkic et al. [34] is used to model PV cells, which are directly connected to one cell of the alkaline water electrolyser model from Section 3.1. Four PV cells (standard industry products, details in

Table A7) are modelled in series to get both devices into a similar operating range. The panels and the electrolyser are both assumed to be isothermal: the panels at 298 K and the electrochemical cell at 323 K. For the irradiance, the time-continuous `Irradiance` model within the *PhotoVoltaics* library is used, which is based on Quaschning [66]. We chose an arbitrary summer day (9 July 2024) and the city of Aachen (in the coordinate form) as input parameter values for the model of the irradiance; the detailed list of the chosen parameter values can be found in Table A7.

To keep this case study simple, we connect the electrolyser directly to the PV cells. Henceforth, there is no DC/DC-Converter nor Maximum Powerpoint Tracking (i.e., a control scheme to maximise solar-cell efficiency) [67]. Subsequently, the operating points of the PV cells and the electrolyser are determined by the intersection of their characteristic, nonlinear *U–I* curves for a given point in time. To be precise, the potential across the electrolyser is equal in value to the potential across the four PV cells, as is the current. From the simulation, the irradiance, the potential across the electrolysis cell, and the power converted are plotted over the time of the day, in Figure 9.

Irradiance accounts for solar radiation without any interference caused by weather or other phenomena. The (electrochemical) cell potential jumps directly at sunrise to 1.5 V and increases then slower and slower until midday, when it reaches its maximum. This is a consequence of the nonlinear behaviour of electrolysers, where the equilibrium potential needs to be exceeded for the electrochemical reactions to occur.

Lastly, the curve of the converted power, mostly, tracks the curve of the solar irradiance. The curve of the converted power is a bit below the irradiance curve at intermediate irradiance levels compared to the maximal irradiance levels. If the ratio of solar irradiation converted to power was constant across the operating range of this setup, these two curves would track each other perfectly; instead, the PV cells are constrained by the electrolyser, showcasing how eCherry can be combined to model the interaction of systems with electrochemical and renewable energy plants. For a more detailed explanation of how the two operating curves influence each other, we refer the reader to Section A.2.

4 | Conclusion and Outlook

In this paper, we have introduced the **E**lectro**che**mical **R**eacto**r** D**y**namics Modelica Library (**eCherry**) and presented its versatility in dynamic modelling of electrochemical reactors. We explain the main structure, how, based on a structured modelling approach and ordered along three domains (electrical, material, and thermal), we aggregate single-domain models into more complex system models. For many electrochemical reactors, a model can be built without writing any code but instead only selecting the appropriate units and connecting these graphically, and setting appropriate parameter values. For more specialised models, the user can either write their own tailored units or use the simple models to build their own units. The acausal modelling approach of eCherry makes it very easy to switch between inputs and outputs for the same simulation.
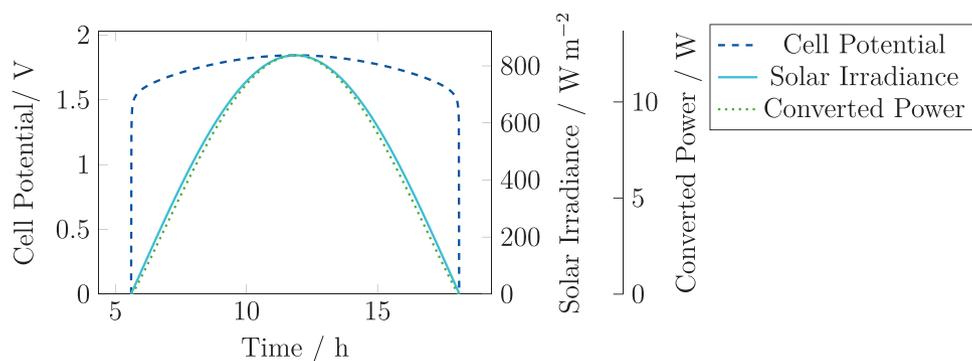
FIGURE 9 | The time series of the solar irradiance, the cell potential, and the power of a combined photovoltaic and electrolyser setup during a typical day without cloud.

We demonstrated a few of the potential systems that can be modelled with eCherry, from modelling electrolysers with one or multiple reactions, investigating heating phenomena, reproducing cyclic voltammetry experiments, or showcasing the interplay of an electrolyser with PVs.

Regarding the limitations of eCherry, the lowest level models follow exclusively a lumped modelling approach; a few selected parts are modelled with a 1D-discretised finite volume approach by aggregating these lumped models, but discretisation in two or three spatial dimensions has led to numerical issues in a few trials. At the moment, we do not have two-phase flow patterns, e.g., slug flow. If information about a specific flow pattern is available, they can be approximated with empirical relationships. Appropriate software tailored to distributed systems, such as COMSOL Multiphysics [68] or OpenFOAM [69], are the better tool in the above cases, e.g., for complex flow when investigating the flow channel design.

Another issue inherent to electrochemical reactors is that the phenomena range in scale from pore diffusion at a few nanometers to plants a few meters in size. Building a mechanistic model that includes the mentioned phenomena leads to scaling issues, which result in stiff differential equations and poor conditioning. This necessitates the need to select the appropriate models and the included phenomena for the (research) question being asked.

Furthermore, the goals of generality and flexibility of eCherry introduce many layers of inheritance and abstractions, which raise the entry barrier. We are trying to mitigate this by tutorials and documentation, of which this paper is one part. Finally and obviously, several unit models from the huge possible space of existing electrochemical setups are still missing.

eCherry is an open-source project that will be updated and expanded with more models, e.g., detailed transport models including migration and the electroneutrality condition and models for gas-diffusion-electrodes as employed in CO_2-electrolysis [70]. Other setups, like rotating-disk electrodes, or membraneless electrolysers [71] could also be potentially added in the future. Lastly, the potential of eCherry lies within applying this general toolbox to more and more specific applications and use cases to be explored, e.g., economic nonlinear model predictive control of electrochemical processes.

Software Availability

The code of eCherry can be found under: https://git.rwth-aachen.de/avt-svt/public/echerry

Author Contributions

Jan F. Pyschik: conceptualisation, methodology, software, validation, formal analysis, investigation, writing–original draft. **Katharina M. Ebeling:** conceptualisation, methodology, software, validation, formal analysis, investigation, writing–original draft. **Georgia Ioanna Prokopou:** conceptualisation, methodology, software, validation, formal analysis, investigation, writing–original draft. **Michael Rix:** conceptualisation, methodology, software, validation, formal analysis, investigation, writing–original draft. **J. Raphael Seidenberg:** conceptualisation, methodology, software, validation, formal analysis, investigation, writing–original draft. **Marius B. Kleiner:** methodology, software, validation, writing–original draft. **Matthias Leitl:** conceptualisation, methodology, writing–review and editing. **Alexander Mitsos:** conceptualisation, writing–review and editing, supervision, funding acquisition. **Dominik Bongartz:** conceptualisation, writing–review and editing, supervision, funding acquisition.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The code for the library introduced in this paper as well as the examples is freely available at https://git.rwth-aachen.de/avt-svt/public/echerry.

References

1. J. Whitehead, P. Newman, J. Whitehead, and K. L. Lim, "Striking the Right Balance: Understanding the Strategic Applications of Hydrogen in Transitioning to a Net Zero Emissions Economy," *Sustainable Earth Reviews* 6, no. 1 (2023): 1.

2. L. T. Biegler, I. E. Grossmann, and A. W. Westerberg, *Systematic Methods of Chemical Process Design*, Prentice-Hall International Series in the Physical and Chemical Engineering Sciences (Upper Saddle River, NJ: Prentice Hall, repr. with corrections edition, 1999).

3. X. Shi, Y. Qian, and S. Yang, "Fluctuation Analysis of a Complementary Wind–Solar Energy System and Integration for Large Scale Hydrogen Production," *ACS Sustainable Chemistry & Engineering* 8, no. 18 (2020): 7097–7110.

4. W. Marquardt, "Trends in Computer-Aided Process Modeling," *Computers & Chemical Engineering* 20, no. 6–7 (1996): 591–609.

5. P. I. Barton and C. C. Pantelides, "Modeling of Combined Discrete/Continuous Processes," *AIChE Journal* 40, no. 6 (1994): 966–979.

6. J. Gesenhues, M. Hein, M. Ketelhut, et al., "Benefits of Object-Oriented Models and Modelichart: Modern Tools and Methods for the Interdisciplinary Research on Smart Biomedical Technology," *Biomedical Engineering* 62, no. 2 (2017): 111–121.

7. D. S. Falcão and A. Pinto, "A Review on Pem Electrolyzer Modelling: Guidelines for Beginners," *Journal of Cleaner Production* 261 (2020): 121184.

8. M. K. Ratib, K. M. Muttaqi, M. R. Islam, D. Sutanto, and A. P. Agalgaonkar, "Electrical Circuit Modeling of Proton Exchange Membrane Electrolyzer: The State-of-the-Art, Current Challenges, and Recommendations," *International Journal of Hydrogen Energy* 49 (2024): 625–645.

9. A. Majumdar, M. Haas, I. Elliot, and S. Nazari, "Control and Control-Oriented Modeling of Pem Water Electrolyzers: A Review," *International Journal of Hydrogen Energy* 48, no. 79 (2023): 30621–30641.

10. S. Zhang, S. B. Beale, U. Reimer, M. Andersson, and W. Lehnert, "Polymer Electrolyte Fuel Cell Modeling—A Comparison of Two Models With Different Levels of Complexity," *International Journal of Hydrogen Energy* 45, no. 38 (2020): 19761–19777.

11. P. Olivier, C. Bourasseau, and P. B. Bouamama, "Low-Temperature Electrolysis System Modelling: A Review," *Renewable and Sustainable Energy Reviews* 78 (2017): 280–300.

12. C. Daoudi and T. Bounahmidi, "Overview of Alkaline Water Electrolysis Modeling," *International Journal of Hydrogen Energy* 49 (2024): 646–667.

13. S. Hu, B. Guo, S. Ding, et al., "A Comprehensive Review of Alkaline Water Electrolysis Mathematical Modeling," *Applied Energy* 327 (2022): 120099.

14. M. Ni, "An Electrochemical Model for Syngas Production by Co-Electrolysis of H_2O and CO_2," *Journal of Power Sources* 202 (2012): 209–216.

15. L. C. Weng, A. T. Bell, and A. Z. Weber, "Modeling Gas-Diffusion Electrodes for CO_2 Reduction," *Physical Chemistry Chemical Physics* 20, no. 25 (2018): 16973–16984.

16. C. Delacourt and J. Newman, "Mathematical Modeling of CO_2 Reduction to CO in Aqueous Electrolytes," *Journal of the Electrochemical Society* 157, no. 12 (2010): B1911.

17. A. R. T. Morrison, V. van Beusekom, M. Ramdin, L. J. P. van den Broeke, T. J. H. Vlugt, and W. de Jong, "Modeling the Electrochemical Conversion of Carbon Dioxide to Formic Acid or Formate at Elevated Pressures," *Journal of The Electrochemical Society* 166, no. 4 (2019): E77–E86.

18. L. Järvinen, P. Puranen, A. Kosonen, et al., "Automized Parametrization of Pem and Alkaline Water Electrolyzer Polarisation Curves," *International Journal of Hydrogen Energy* 47, no. 75 (2022): 31985–32003.

19. F. J. Folgado, I. González, and A. J. Calderón, "Simulation Platform for the Assessment of Pem Electrolyzer Models Oriented to Implement Digital Replicas," *Energy Conversion and Management* 267 (2022): 115917.

20. S. Zhang, S. Hess, H. Marschall, U. Reimer, S. Beale, and W. Lehnert, "Openfuelcell2: A New Computational Tool for Fuel Cells, Electrolyzers, and Other Electrochemical Devices and Processes," *Computer Physics Communications* 298 (2024): 109092.

21. S. B. Beale, H.-W. Choi, J. G. Pharoah, H. K. Roth, H. Jasak, and D. H.s Jeon, "Open-Source Computational Model of a Solid Oxide Fuel Cell," *Computer Physics Communications* 200 (2016): 15–26.

22. V. Novaresio, M. García-Camprubí, S. Izquierdo, P. Asinari, and N. Fueyo, "An Open-Source Library for the Numerical Modeling of Mass-Transfer in Solid Oxide Fuel Cells," *Computer Physics Communications* 183, no. 1 (2012): 125–146.

23. B. Kreitz, J. Brauns, G. D. Wehinger, and T. Turek, "Modeling the Dynamic Power–to–Gas Process: Coupling Electrolysis With CO_2 Methanation," *Chemie Ingenieur Technik* 92, no. 12 (2020): 1992–1997.

24. J. Webster and C. Bode, "Implementation of a Non-Discretized Multiphysics Pem Electrolyzer Model in Modelica," in *Proceedings of the 13th International Modelica Conference, Regensburg, Germany, March 4–6, 2019*, Linköping Electronic Conference Proceedings (Linköing University Electronic Press, 2019), 833–840.

25. J. S. Kim, S. M. Bragg-Sitton, and R. D. Boardman, *Status Report on the High-Temperature Steam Electrolysis Plant Model Developed in the Modelica Framework (fy17)*, Technical Report (2017).

26. A. Koksharov, Arnulf Latz, and Thomas Jahnke, "Neopard-x: Multidimensional Highly Efficient and Scalable Modular Framework for Modelling of Electrolysers and Fuel Cells," *Electrochimica Acta* 495 (2024): 144482.

27. M. Matejak, M. Tribula, F. Ježek, and J. Kofranek, "Free Modelica Library for Chemical and Electrochemical Processes," in *Proceedings of the 11th International Modelica Conference, Versailles, France, September 21-23, 2015*, Linköping Electronic Conference Proceedings (Linköping University Electronic Press, 2015), 359–366.

28. A. M. Bremen, K. M. Ebeling, V. Schulte, J. Pavšek, and A. Mitsos, "Dynamic Modeling of Aqueous Electrolyte Systems in Modelica," *Computers & Chemical Engineering* 166 (2022): 107968.

29. D. E. Keyes, L. C. McInnes, C. Woodward, et al., "Multiphysics Simulations: Challenges and Opportunities," *The International Journal of High Performance Computing Applications* 27, no. 1 (2013): 4–83.

30. D. Amirante, V. Ganine, N. J. Hills, and P. Adami, "A Coupling Framework for Multi-Domain Modelling and Multi-Physics Simulations," *Entropy* 23, no. 6 (2021): 758.

31. M. Otter, "Multi-Domain Modeling and Simulation," in *Encyclopedia of Systems and Control*, eds. J. Baillieul and T. Samad (Cham: Springer, 2021), 1367–1379.

32. M. Tiller, "Multi-Domain Modeling," in *Introduction to Physical Modeling with Modelica*, The Kluwer International Series in Engineering and Computer Science (Boston, MA: Springer, 2001), 231–253.

33. D. Britz and J. Strutwolf, *Digital Simulation in Electrochemistry* (Cham: Springer, 2016).

34. J. Brkic, M. Ceran, M. Elmoghazy, R. Kavlak, A. Haumer, and C. Kral, "Open Source Photovoltaics Library for Systemic Investigations," in *Proceedings of the 13th International Modelica Conference, Regensburg, Germany, March 4–6, 2019*, Linköping Electronic Conference Proceedings (Linköing University Electronic Press, 2019), 41–50.

35. P. Fritzson and V. Engelson, "Modelica—A Unified Object-Oriented Language for System Modeling and Simulation," in *ECOOP'98-Object-oriented programming*, ed. E. Jul, Lecture Notes in Computer Science (Berlin: Springer, 1998), 67–90.

36. Modelica Association, Modelica—A Unified Object-Oriented Language for Systems Modeling Language Specification: Version 3.6 (2023).

37. D. Systèmes, *Dymola-dassault systèmes*, accessed Jul 30, 2024, https://www.3ds.com/.

38. W. Marquardt, "An Object-Oriented Representation of Structured Process Models," *Computers & Chemical Engineering* 16 (1992): S329–S336.

39. G. J. Klir, *Architecture of Systems Problem Solving* (New York, NY: Springer, 1985).

40. E. J. F. Dickinson and G. Hinds, "The Butler–Volmer Equation for Polymer Electrolyte Membrane Fuel Cell (pemfc) Electrode Kinetics: A Critical Discussion," *Journal of The Electrochemical Society* 166, no. 4 (2019): F221–F231.

41. F. Kreith, ed, *The CRC Handbook of Mechanical Engineering*, The Mechanical Engineering Handbook Series (Boca Raton, FL: CRC Press, 1998).

42. J. Newman and K. E. Thomas-Alyea, *Electrochemical Systems*, 3rd ed. (Somerset: Wiley, 2012).

43. S. N. Boroujeni, X. Liang, B. Maribo-Mogensen, and G. M. Kontogeorgis, "Comparison of Models for the Prediction of the Electrical Conductivity of Electrolyte Solutions," *Industrial & Engineering Chemistry Research* 61, no. 8 (2022): 3168–3185.

44. R. Gilliam, J. Graydon, D. Kirk, and S. Thorpe, "A Review of Specific Conductivities of Potassium Hydroxide Solutions for Various Concentrations and Temperatures," *International Journal of Hydrogen Energy* 32, no. 3 (2007): 359–364.

45. D. A. G. Bruggeman, "Berechnung Verschiedener Physikalischer Konstanten Von Heterogenen Substanzen. ii. Dielektrizitätskonstanten und Leitfähigkeiten Von Vielkristallen Der Nichtregulären Systeme," *Annalen der Physik* 417, no. 7 (1936): 645–672.

46. R. Krishna and J. A. Wesselingh, "The Maxwell–Stefan Approach to Mass Transfer," *Chemical Engineering Science* 52, no. 6 (1997): 861–911.

47. O. Walz, C. Marks, J. Viell, and A. Mitsos, "Systematic Approach for Modeling Reaction Networks Involving Equilibrium and Kinetically-Limited Reaction Steps," *Computers & Chemical Engineering* 98 (2017): 143–153.

48. E. J. F. Dickinson and A. J. Wain, "The Butler–Volmer Equation in Electrochemical Theory: Origins, Value, and Practical Application," *Journal of Electroanalytical Chemistry* 872 (2020): 114145.

49. R. Guidelli, R. G. Compton, J. M. Feliu, et al., "Defining the Transfer Coefficient in Electrochemistry: An Assessment (IUPAC Technical Report)," *Pure and Applied Chemistry* 86, no. 2 (2014): 245–258.

50. R. Guidelli, R. G. Compton, J. M. Feliu, et al., "Definition of the Transfer Coefficient in Electrochemistry (IUPAC Recommendations 2014)," *Pure and Applied Chemistry* 86, no. 2 (2014): 259–262.

51. J. R. Seidenberg, A. Mitsos, and D. Bongartz, *Interpreting Concentration and Activation Overpotentials in Electrochemical Systems: A Critical Discussion* (2024), Published ahead of print, https://doi.org/10.1149/osf.io/ctkf6.

52. E. R. Cohen, E. R. Cohen, and I. Mills, eds., *Quantities, Units and Symbols in Physical Chemistry* (RSC Publication, Cambridge, 3rd ed., 2008).

53. T. E. Springer, T. A., Zawodzinski, and S. Gottesfeld, "Polymer Electrolyte Fuel Cell," *Journal of The Electrochemical Society* 138, no. 8 (1991): 498–501.

54. J. Brauns, J. Schönebeck, M. R. Kraglund, et al., "Evaluation of Diaphragms and Membranes as Separators for Alkaline Water Electrolysis," *Journal of The Electrochemical Society* 168, no. 1 (2021): 014510.

55. M. Hammoudi, C. Henao, K. Agbossou, Y. Dubé, and M. L. Doumbia, "New Multi-Physics Approach for Modelling and Design of Alkaline Electrolyzers," *International Journal of Hydrogen Energy* 37, no. 19 (2012): 13895–13913.

56. C. Henao, K. Agbossou, M. Hammoudi, Y. Dubé, and A. Cardenas, "Simulation Tool Based on a Physics Model and an Electrical Analogy for an Alkaline Electrolyser," *Journal of Power Sources* 250, no. 7 (2014): 58–67.

57. L. C. Brée, M. Wessling, and A. Mitsos, "Modular Modeling of Electrochemical Reactors: Comparison of CO_2-Electolyzers," *Computers & Chemical Engineering* 139 (2020): 106890.

58. G. Sakas, A. Ibáñez-Rioja, V. Ruuskanen, A. Kosonen, J. Ahola, and O. Bergmann, "Dynamic Energy and Mass Balance Model for an Industrial Alkaline Water Electrolyzer Plant Process," *International Journal of Hydrogen Energy* 47, no. 7 (2022): 4328–4345.

59. N. Elgrishi, K. J. Rountree, B. D. McCarthy, E. S. Rountree, T. T. Eisenhart, and J. L. Dempsey, "A Practical Beginner's Guide to Cyclic Voltammetry," *Journal of Chemical Education* 95, no. 2 (2018): 197–206.

60. C. H. Hamann, A. Hamnett, and W. Vielstich, *Electrochemistry* (Wiley-VCH, 2007).

61. M. H. Pournaghi-Azar and R. Ojani, "Electrode Kinetic Parameters of the Ferrocene Oxidation at Platinum, Gold and Glassy Carbon Electrodes in Chloroform," *Electrochimica Acta* 39, no. 7 (1994): 953–955.

62. L. Göke, J. Weibezahn, and M. Kendziorski, "How Flexible Electrification can Integrate Fluctuating Renewables?," *Energy* 278 (2023): 127832.

63. A. E. Samani, A. D'Amicis, J. D. M. de Kooning, D. Bozalakov, P. Silva, and L. Vandevelde, "Grid Balancing With a Large–Scale Electrolyser Providing Primary Reserve," *IET Renewable Power Generation* 14, no. 16 (2020): 3070–3078.

64. R. Cozzolino and G. Bella, "A Review of Electrolyzer-Based Systems Providing Grid Ancillary Services: Current Status, Market, Challenges and Future Directions," *Frontiers in Energy Research* 12 (2024): 1358333.

65. R. Franke and H. Wiesmann, "Flexible Modeling of Electrical Power Systems—the Modelica Powersystems Library," in *Proceedings of the 10th International Modelica Conference, March 10–12, 2014, Lund, Sweden*, Linköping Electronic Conference Proceedings (Linköping University Electronic Press, 2014), 515–522.

66. V. Quaschning, *Regenerative Energiesysteme: Technologie -Berechnung-Simulation* (München: Hanser, 7th ed. 2011).

67. A. N. A. Ali, M. H. Saied, M. Z. Mostafa, and T. M. Abdel-Moneim, "A Survey of Maximum ppt Techniques of pv Systems," in *2012 IEEE Energytech* (Piscataway, NJ: IEEE, 2012), 1–17.

68. COMSOL AB. Comsol multiphysics v. 6.2. Reference Manual (2023).

69. H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, "A Tensorial Approach to Computational Continuum Mechanics Using Object-Oriented Techniques," *Computers in Physics* 12, no. 6 (1998): 620–631.

70. M. Heßelmann, B. C. Bräsel, R. G. Keller, and M. Wessling, "Simulation–Based Guidance for Improving CO_2 Reduction on Silver Gas Diffusion Electrodes," *Electrochemical Science Advances* 3, no. 1 (2023): 2100160.

71. X. Pang, S. Das, J. T. Davis, and D. V. Esposito, "Membraneless Electrolyzers for Low-Cost Hydrogen Production," *ECS Meeting Abstracts* MA2020-01, no. 37 (2020): 1587.

72. E. Henningsson, H. Olsson, and L. Vanfretti, "Dae Solvers for Large-Scale Hybrid Models," in *Linköping Electronic Conference Proceedings* (Linköing University Electronic Press, 2019).

73. C. C. Pantelides and P. I. Barton, "Equation-Oriented Dynamic Simulation Current Status and Future Perspectives," *Computers & Chemical Engineering* no. 17 (1993): 263–285.

74. L. J. J. Janssen and J. G. Hoogland, "The Effect of Electrolytically Evolved Gas Bubbles on the Thickness of the Diffusion Layer—ii," *Electrochimica Acta* 18, no. 8 (1973): 543–550.

75. P. Vermeiren, W. Adriansens, J. P. Moreels, and R. Leysen, "Evaluation of the Zirfon Separator for use in Alkaline Water Electrolysis and Ni − H_2 Batteries," *International Journal of Hydrogen Energy* 23, no. 5 (1998): 321–324.

Appendix A

A.1 | Terminology

To help with the understandability of the library, we provide a short summary of jargon terms that are either Modelica specific or introduced by us to describe eCherry (Table A1)

A.2 | Photovoltaics and Electrolysers

To understand the relationship between an electrolyser and photovoltaics, we take a look at their characteristic current–voltage graphs, as often done in the literature. These two nonlinear curves are graphs of steady-state working points. Within eCherry, the current–voltage graph is a result of the interlinked multiphysical phenomena within the electrolyser.

For a photovoltaic cell, the plot ranges from a maximum current, with zero potential (known as open circuit voltage) to a maximum potential at zero current (known as short-circuit current). Given that power is the product of voltage and current and the extreme value theorem, the maximal power output of an photovoltaic cell lies in-between those two points as is visualised in Figure A1.

It showcases how for a PV cell there is an optimal operating point maximising the power output. This maximum depends on the irradiance, as the extension of this curve is influenced by the same.

If we were simulating the electrolyser and the PV cells conventionally, we would simulate each individually and have to iterate for each point in time until the operating points of voltage and current match up.
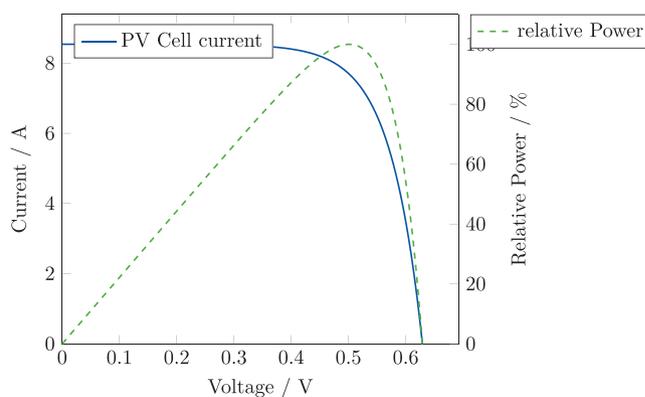

FIGURE A1 | Characteristic current–voltage plot for one photovoltaic cell with a norm irradiance of 1000 W m^{-2} as used in Section 3.5. Additionally, the relative power output, scaled to its maximum, is plotted over the voltage.

When a simulation is run, Modelica and solves the system of all the equations describing the model. In Figure A2, the current–voltage graph for the electrolyser and for the four PV cells (in-series) are drawn for two exemplary cases of an irradiance of 500 and 1000 W m^{-2}.

TABLE A1 | (Modelica) terminology.

| Term	Explanation
variable	Quantity whose (time-dependent) value is determined in the simulation.
flow variable	Describes the movement of a conserved quantity; connection imposes that the sum of all flow variables is zero.
potential variable	Difference describes a driving force; connection sets all potential variables equal.
connector	Comprises flow and potential variables that enable the transfer of variable values between modular models.
connect	The act of transferring variable values between models via connectors.
model	Modelica file within the library, collecting parameters, variables, and equations.
base model	Model with most but not all equations of a unit model to avoid duplicated code.
unit model	Ready-to-use; can be graphically visualised and dragged and dropped in the GUI to build a system model.
example model	Ready-to-simulate; includes all required parameter and initial values.
system model	An example model that represents a real system (e.g., electrolszer, chemical plant).
lowest-level model	A model that is impossible or not useful to be splitted further into more than one model.
lower-level model	Combines two or a few more lowest-level models.
higher-level model	Combines several lower-level models.
domain	Collection of models that include either material balance, energy balance or Kirchhoffs-law equations.
single-domain	Consists of exactly one domain.
multi-domain	Consists of more than one domain.
electrical domain	Models only include equations for current–voltage characteristics.
material domain	Models only include equations for material conservation and transport.
thermal domain	Models only include equations for energy conservation and transport.
package	Directory where Modelica code is stored.
record	Modelica structure to collect parameters and their values.
library	Collection of packages containing models, example models, and data.
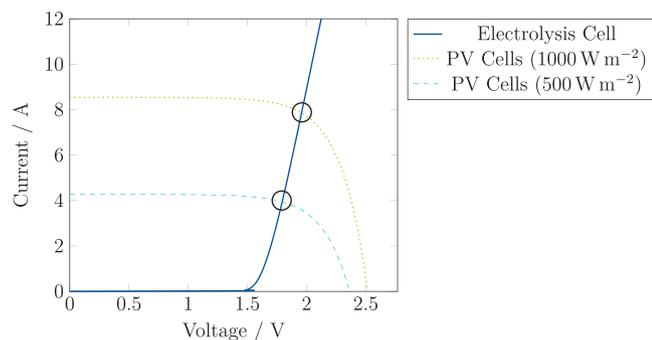
FIGURE A2 | Characteristic current–voltage plots for the alkaline water electrolysis and PV cells at two different irradiance levels. The intersections which describe the operating points of the combined system are denoted by circles.

The two circles highlight the operating points of the entire system for the two different irradiances. In industrial application, this source of inefficieny is avoided by using a DC/DC-converter with an inbuilt optimiser (called maximum-power-point-tracking), as seen by Ali et al. [67]. This case study is meant to show a non-trivial, but still quite simple, example of how an electrolyser can be simulated together with other electric devices.

A.3 | Runtimes

The time one simulation takes is usually in the order of a few seconds, as is expected from modern tools like Dymola, cf., Henningsson et al. [72], due to the development of efficient solvers for DAE systems, e.g., Pantelides and Barton [73]. Thus far, spatial discretisation appears to be the main aspect leading to high runtimes. All simulations were executed on a Dell Latitude 5421 Laptop (CPU: 11th Gen Intel(R) Core(TM) i7-11850H @ 2.50GHz). In Table A2 the recorded runtimes of the illustrative examples from Section 3 are given. The difference between total runtime (t_{total}) and the time for integration($t_{integration}$) is mainly caused by the time it takes to compile the model.

To investigate the influence of the amount of discretised volumes on the runtime performance, we simulated the cyclic voltammetry example with varying number of finite volumes but the same thickness of the boundary layer ($100\,\mu$m). The results are presented in Figure A3.

As expected, the computational effort increases superlinearly with higher granularity, but overall is very manageable. For $n = 100$ the runtime

TABLE A2 | The runtimes of the five examples.

Example	t_{total} / s	$t_{integration}$ / s
Alkaline water electrolyser	1.9	0.042
Multiple electrochemical reactions	2.8	0.028
Electrolyser cold-start	3.2	0.035
Cyclic voltammetry	781	775
Coupling electrolysis and photovoltaics	9.2	7.26

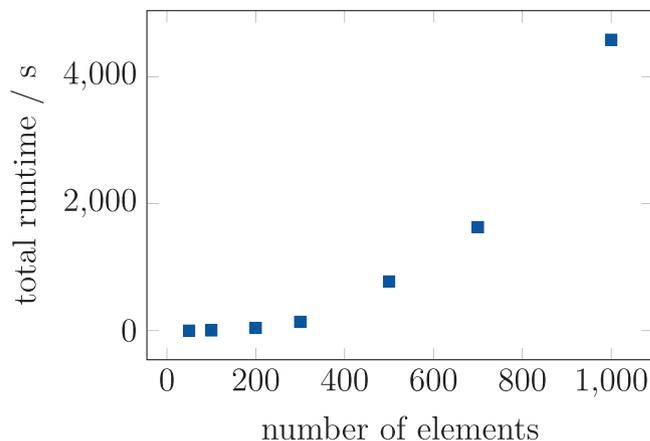*Note:* Shown are the total runtime and the time the numerical integration took.

FIGURE A3 | The total runtime of the cyclic voltammetry example plotted over the number of discretised volumes.

is just 5.3 s and this discretisation appears acceptable, especially considering the small dimensions. Note that for $n = 1000$ and a thickness of the boundary layer of $100\,\mu$m [74], each discretised volume would be only 100 nm thick, which is excessively small: if such small length-scales are relevant even the basic continuum hypothesis needs to be examined.

A.4 | Further Parameters

The major parameters and their values for the case studies in Section 3 can be found here (Tables A3–A7).

TABLE A3 | Parameters and their values used in the simulation of the Stuart electrolyser according to Hammoudi et al. [55] and Henao et al. [56].

Parameter	Description	Value	Unit	Source
T	Temperature	323.15	K	[55]
p	Pressure	1	bar	[55]
$\alpha_{a,\text{OER}}$	Anodic transfer coefficient for O_2 evolution at anode	0.748[a]	—	[55]
$j_{0,\text{ref,OER}}$	Exch. current density for O_2 evolution at anode	0.038	A m^{-2}	[56]
$\alpha_{c,\text{HER}}$	Cathodic transfer coefficient for H_2 evolution at cathode	0.848[a]	—	[55]
$j_{0,\text{ref,HER}}$	Exch. current density for H_2 evolution at cathode	$9.4 \cdot 10^2$	A m^{-2}	[56]
A	Active area	0.03	m^2	[55]
x_D	Diaphragm thickness	$0.5 \cdot 10^{-3}$	m	[56]
x_gap	Electrode diaphragm gap	$1.25 \cdot 10^{-3}$	m	[55]
$\dot{V}$	Total electrolyte flow	500	mLmin^{-1}	[55]
κ_D	Diaphragm conductivity	27.1	Sm^{-1}	[75]

[a]Taking into account the z used in front of transfer coefficient by Hammoudi et al. [55]

TABLE A4 | Parameters and their values used in the simulation of the CO_2 electrolyser based on Brée et al. [57].

Parameter	Description	Value	Unit	Source
T	Temperature	293.15	K	[57]
$\alpha_{a,\text{OER}}$	Anodic transfer coefficient of O_2 evolution at anode	0.89	—	[57]
$\alpha_{c,\text{OER}}$	Cathodic transfer coefficient of O_2 evolution at anode	$1-\alpha_{a,\text{OER}}$	–	[57]
$j_{0,\text{ref,OER}}$	Exch. current density of O_2 evolution at anode	7.16	$A\,m^{-2}$	[57]
$\alpha_{c,\text{HER}}$	Cathodic transfer coefficient of H_2 reduction at cathode	0.66	—	Assumption
$\alpha_{a,\text{HER}}$	Anodic transfer coefficient of H_2 reduction at cathode	$1-\alpha_{c,\text{HER}}$	—	[57]
$j_{0,\text{ref,HER}}$	Exch. current density of H_2 reduction at cathode	$19.63 \cdot 10^{-7}$	$A\,m^{-2}$	Assumption[a]
$\alpha_{c,\text{CO}_2\text{RR}}$	Cathodic transfer coefficient of CO_2 reduction at cathode	0.75	—	[57]
$\alpha_{a,\text{CO}_2\text{RR}}$	Anodic transfer coefficient of CO_2 reduction at cathode	$1-\alpha_{c,\text{CO}_2\text{RR}}$	—	[57]
$j_{0,\text{ref,CO}_2\text{RR}}$	Exch. current density of CO_2 reduction at cathode	$1.65 \cdot 10^{-2}$	$A\,m^{-2}$	Assumption[a]
A	Active area	4.5	cm^2	[57]
$\dot{V}$	Total electrolyte flow	$1 \cdot 10^{-4}$	$m^3 s^{-1}$	[57]
x_{memb}	Membrane thickness	$115 \cdot 10^{-6}$	m	[57]
κ_{memb}	Membrane conductivity	9.3	Sm^{-1}	[57]
κ_{ele}	Electrolyte conductivity	9.3	Sm^{-1}	[57]

[a]Assumption to get reasonable Faradaic efficiencies.

TABLE A5 | Parameters and their values used in the simulation of the heat-up behaviour example, all values taken from Sakas et al. [58].

Parameter	Description	Value	Unit
$j_{\text{cell,max}}$	Current density	1717	$A\,m^{-2}$
T_0	Initial stack temperature	303.15	K
T_{amb}	Ambient temperature	303.05	K
p	Pressure	16	bar
c_p	Molar heat capacity Hydrogen	28.82	$J kg^{-1} K^{-1}$
	Molar heat capacity Oxygen	29.38	$J kg^{-1} K^{-1}$
	Molar heat capacity other components	75	$J kg^{-1} K^{-1}$
ρ	Density	1280	$kg\,m^{-3}$
A	Active area	2.66	m^2
$X_{\text{anolyte,catholyte}}$	Width of anolyte/catholyte	4.75	mm
$X_{\text{anode,cathode}}$	Thickness of anode/cathode	3.25	mm
X_{membrane}	Thickness of membrane	0.5	mm
M	Molality of the solution	5.941	$mol kg^{-1}$
α_1	Model parameter	0.8	Ωcm^2
α_2	Model parameter	-0.00763	$\Omega cm^2\,°C^{-1}$
s	Model parameter	0.1795	V
β_1	Model parameter	20	$cm^2\,A^{-1}$
β_2	Model parameter	0.1	$cm^2\,°C A^{-1}$
β_3	Model parameter	$3.5 \cdot 10^5$	$cm^2\,°C^2\,A^{-1}$

TABLE A6 | Parameters and their values used in the simulation of the cyclic voltammetry.

Parameter	Description	Value	Unit	Source
ΔE	Applied potential delta	0.7 (-0.4 to 0.3)	V	Assumption
T	Temperature	303	K	[59]
$\alpha_{a,\text{FCOR}}$	Anodic transfer coefficient for Fc oxidation	0.33	—	[61]
$\alpha_{c,\text{FCOR}}$	Cathodic transfer coefficient for Fc oxidation	$1 - \alpha_{an}$	—	Assumption
$j_{0,\text{ref,FCOR}}$	Exch. current density for Fc oxidation	21200[a]	Am^{-2}	[61]
D_{Fc}	Diffusion coefficient of Fc	$2.7 \cdot 10^{-9}$	$\text{m}^2\,\text{s}^{-1}$	[59]
D_{Fc^+}	Diffusion coefficient Fc^+	$2.9 \cdot 10^{-9}$	$\text{m}^2\,\text{s}^{-1}$	[59]
A_{model}	Active Area of model (for numerical stability)	10	m^2	Assumption
A_{exp}	Active Area for calculation of real current	$7.065 \cdot 10^{-6}$	m^2	[59]

[a]Calculated as $j_{0,\text{ref}} = k_0 \cdot F \cdot C = 2.2 \cdot 10^{-2}\,\text{cms}^{-1} \cdot 1 \cdot 10^{-2}\,\text{mcm}^{-1} \cdot 96485\,\text{Cmol}^{-1} \cdot 1 \cdot 10^3\,\text{mol/m}^3 \approx 21200\,\text{A/m}^2$.

TABLE A7 | Parameters and their values used in the simulation of the combined photovoltaics and electrolyser case study.

Parameter	Description	Value	Unit
Latitude	—	50.775555	°
Longitude	—	6.083611	°
Date	—	2024-07-09	—
gamma	Angle of PV cell from horizontal plane	10	°
T_0	Working temperature	298	K
VmpCref	Reference maximum power current	0.5	V
Impref	Reference maximum cell voltage	7.71	A
$N_{\text{cell,series}}$	PV cells in series	4	—
$A_{\text{Electrolyser}}$	Surface area of electrodes	25	m^2

For the PV cells, the data from the commercially available *SHARP NU monocrystalline SI cell 185W* are used.