Free Modelica Library of Chemical and Electrochemical Processes

Chemical 1.1.0

Marek Mateják[1], Martin Tribula[1], Filip Ježek[2], Jiří Kofránek[1,2]

[1]Institute of Pathological Physiology, 1st Faculty of Medicine, Charles University in Prague
U Nemocnice 5, Prague 2, 128 53, Czech Republic
[2]Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague,
Technicka 2, Prague 6

marek@matfyz.cz

Abstract

A new, free Modelica library for electrochemical processes has been released - accessible as “Chemical” at https://www.modelica.org/libraries. It is based on equilibrating the electrochemical potentials of the substances involved, following the modern theories of physical chemistry. It dynamically solves the chemical equilibration of homogeneous chemical solutions with fully thermodynamic states, supported also through thermal, mechanical, electrical and fluid components of Modelica Standard Library 3.2.1. Even the complex processes can be composed from only a few base components, such as a component for the chemical solution, a component for the chemical substance or a component for the chemical reaction. Behind these components are fundamental laws of thermodynamics and physical chemistry. The library was designed to be very intuitive and easy to use. This paper shows how the library can be used to implement the examples of a lead-acid battery, a hydrogen burning and a chloride shift of human red blood cells.

Keywords: Modelica library, physical chemistry, thermodynamics equilibria, electrochemical potential, electrochemical cell, internal energy, semipermeable membrane

1 Introduction

The content for the chemical library comes from Physiolibrary, a library for physiological calculations [1, 2]. We used Physiolibrary to implement the most extensive model of human physiology in 2010: HumMod [3-5]. We named our extended model Physiomodel, and we have continued to extend it at more detailed microscopic and chemical levels. The macroscopic processes and regulations of human physiology are already validated by experiments on animals and humans [6]. However, the chemical processes of the models were (until now) conceived in terms of black boxes with inputs and outputs defined more by empirical relationships than by strict physical theory. Focusing on empirical behavior meant that expectations of the elementary processes were well formed. This chemical library allows us to move different substances in different directions across a membrane at the same time, which was not possible when using, for example, the Modelica.Fluid package [7] because stream constructs move all substances together only in the direction of the main solution stream. However having a set of substance connectors (Table 1) there is possible to change each substance separately just by setting its molar flow.

Table 1, Connector for substance: SubstancePort ■□

nonflow	flow
Electrochemical potential of the substance [J/mol]	*Molar flow of the substance* [mol/s]

In the Chemical library, we carefully selected only the fundamental definitions from physical chemistry and thermodynamics to derive other known chemical relations [8]. For example, physical chemistry defines an electrochemical potential $\overline{\boldsymbol{\mu_j}}$ (Eq.1) for each chemical substance $\boldsymbol{j}$ in a homogeneous chemical solution as the composition of a relative molar energy of pure substance $\boldsymbol{\mu_j^o}$ (typically tabulated as free molar Gibbs energy of formation), a chemical dissolution component of molar energy $\mathrm{R} \cdot T \cdot \ln(a_j)$ (reflecting the mole-fraction based activity of the substance $\boldsymbol{a_j}$ in the solution) and an electrical component of the molar energy $\mathrm{F} \cdot z_j \cdot \varphi$ (for substances with charge number $\boldsymbol{z_j}$ in the solution with non-zero electrical potential $\boldsymbol{\varphi}$), where $\boldsymbol{T}$ is temperature, **R** is gas constant and **F** is Faraday’s constant (Eq.1).

$$\overline{\boldsymbol{\mu_j}} = \boldsymbol{\mu_j^o} + \mathbf{R} \cdot \boldsymbol{T} \cdot \ln\left(\boldsymbol{a_j}\right) + \mathbf{F} \cdot \boldsymbol{z_j} \cdot \boldsymbol{\varphi} \qquad \text{Eq. 1}$$

The relative energy of the pure substance μ_j^o must be compatible with all tabulated equilibrium coefficients: for example, equilibrium coefficients of chemical reactions (as expressed by the free Gibbs energy of the reaction), Henry’s coefficient for gas dissolution equilibrium, Raoult’s vapor pressure equilibrium, standard voltages of redox reactions and so on. These known relations do not need to be explicitly written in code because they are the results of algebraic manipulation of the implemented relations, as we mathematically proved during development. Therefore, in this way the Chemical library married chemical, osmotic, thermal, electrical, mechanical and fluid domains. Usage of the library has been very simplified, because it is typically possible to build many types of reactions with few chemical substances - having a set of already defined chemical substances allows automatic calculation of equilibrium coefficients of their chemical processes. The principles that apply to these free Gibbs energies of substances can also be applied to free heat energies (free enthalpies) because the same relation — called Hess' law — exists between free enthalpy of chemical processes and relative (free) enthalpies of substances which are typically tabulated as free molar enthalpies of formation. Therefore, the user does not even need to set the value of the heat consumed or released from the chemical process, since this heat energy is automatically derived from the substance definitions.

The other problem with the old Physiolibrary approach is that it does not automatically calculate the membrane equilibria for electrically charged substances. The very specific blocks for calculating the Donnan's equilibria [9] at the glomerular membrane were created to reach expected concentrations of electrolytes in semipermeable membranes. The chemical library offered can automatically solve the Donnan's equilibria of a semipermeable membrane, together with the Nernst membrane potential, as a consequence of the equilibrated electrochemical potentials of the permeable substances.

After these electrochemical processes in a cellular membrane was married with chemical reactions, we realized that the relations are general enough to calculate phase changes, gas solubility, electrochemical cells and other known chemical processes as described in physical chemistry textbooks, such as [8]. The result is a library that allows us to create any type of chemical reaction, in any type of homogenous chemical solution, in any conditions. We made it in one hand with thermodynamics and physical chemistry relations behind. In Modelica, the selected base definitions from this theoretical approach can be directly rewritten to the code in their natural mathematical forms, which significantly simplify the implementation.

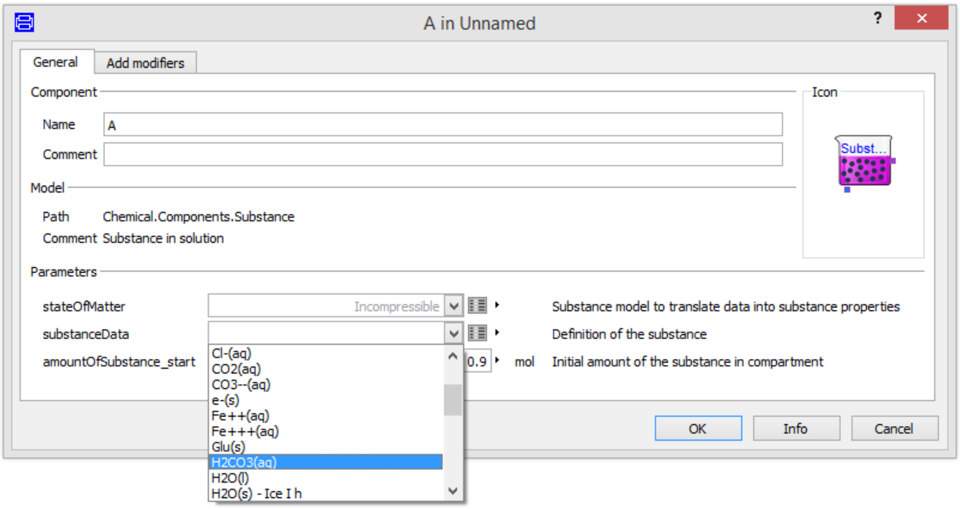
The Chemical library is freely available at https://github.com/MarekMatejak/Chemical and is meant to become a part of Modelica Standard Library.

The model is partially documented directly in the code, more detailed description of the usage, including this article and underlying principles is to be found in attached documents in Documentation folder. This paper shows the main principle and usability of the library on three simple examples.

2 Chemical Substance

The Chemical library in version 1.1.0 contains two basic states of matter: ideal gas and incompressible substance. However, the user can easily (re)define their own state of matter by inserting the correct expressions for the pure substance activity coefficient, molar volume, molar entropy and molar enthalpy, based on the current solution state (temperature, pressure, electric potential and ionic strength) and the substance data. The object-oriented design allows users to define the substance data record as part of the state of matter package. Users can select substance parameters according to the state of matter, redefining the getter functions of substance properties.

Our examples work with ideal gases in case of all gaseous substance and incompressible state of matter in case of liquid or solid. The definition data are the molar mass of the substance, the number of charges of



Name A

Comment

Path Chemical.Components.Substance

Comment Substance in solution

stateOfMatter Incompressible Substance model to translate data into substance properties

substanceData Definition of the substance

amountOfSubstance_start 0.9 mol Initial amount of the substance in compartment

Cl-(aq)
CO2(aq)
CO3--(aq)
e-(s)
Fe++(aq)
Fe+++(aq)
Glu(s)
H2CO3(aq)
H2O(l)
H2O(s) - Ice I h

Figure 1. Setting of the predefined chemical substance, where (s) = solid phase, (aq) = dissolved in water, (g) = gas phase and (l) = liquid phase.

the substance, the molar heat capacity of the substance at a constant pressure, free formation enthalpy, free formation Gibbs energy and density (if incompressible) — all at a temperature of 25°C and pressure 1 bar. Since these parameters are usually recorded in chemical tables at this standard conditions. In this manner, more than 35 real chemical substances in the example package of this chemical library have already been defined. The usage of these predefined substances' data is very simple. In the parameter dialog of the chemical substance, the correct record with this data can be selected, as shown in Figure 1.

This setting is typically the most important setting of each chemical model. All equilibrium coefficients, standard voltages, dissolution coefficients, saturated vapor pressures etc., are automatically solved using these substance data. As a result, for example, the chemical reaction component only needs to define the stoichiometry coefficients, and the connected substances reach equilibrium at the correct equilibrium coefficient.

As a result of fundamental relations, the solution of chemical substances contains enthalpy, entropy and internal energy. These properties can be represented also as Media of MSL 3.2 (e.g. Interfaces.SimpleChemicalMedium). Having solution as homogenous mixture of one state of matter there is an option to use the Fluid connectors and components of MSL 3.2 using Chemical library component named Components.FluidAdapter. The FluidAdapter can connect each substance of the solution with the fluid port, which represent the stream of the whole solution (e.g. Examples.FluidAdapter2).

3 Example of the Lead-Acid Battery

The lead-acid electrochemical cells are characterized by two chemical reactions:

$$PbO_2 + HSO_4^- + 3\ H^+ + 2\ e^- \leftrightarrow PbSO_4 + 2\ H_2O \qquad (1)$$

$$Pb + HSO_4^- \leftrightarrow PbSO_4 + H^+ + 2\ e^- \qquad (2)$$

The building of one cell of a lead-acid battery starts with the definition of three solutions: two for the lead electrodes and one for the liquid-acid solution (Figure 2A). This can be done by dragging and dropping the library class 'Components.Solution' into the diagram. We called the first instance "cathode", the second "solution" and the last "anode". We set the parameter 'ElectricalGround' as "false" for all of these solutions in order to attain the possibility of non-zero voltages. Now we can specify the chemical substances inside the

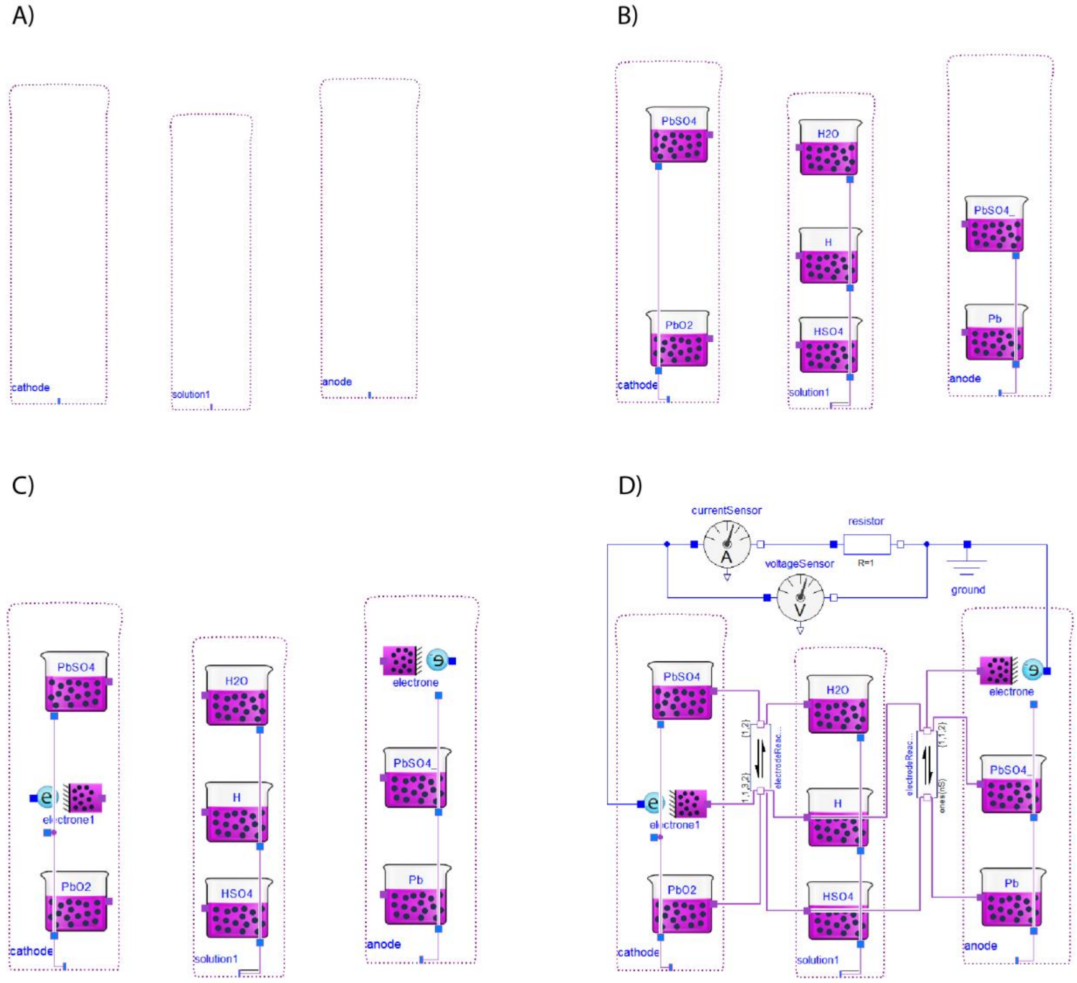


Figure 2. The building of one electro-chemical cell of a lead-acid battery in four steps: A) adding chemical solutions, B) adding chemical substances, C) adding electron transfers and D) adding chemical reactions.

chemical solutions. We drag and drop the library class 'Components.Substance' into the "solution" as chemical substances (Figure 2B). H_2O(liquid), H^+(aqueous) and HSO_4^-(aqueous) representing the liquid aqueous solution of sulfuric acid. $PbSO_4$(solid) and PbO_2(solid) are placed in the "cathode", representing the elements of the positive electrode. The substances Pb(solid) and a$PbSO_4$(solid) are placed into the "anode", representing the elements of the negative electrode. All of these substances must be given unique names (e.g., "PbSO4" for the cathode and "aPbSO4" for the anode), because the Modelica language does not support two instances with the same name in a single class.

As mentioned above, the appropriate substance data for all these substances must be selected as predefined parametric records, e.g., 'Examples.Substances.Water_liquid', '.Lead_solid', '.Lead_dioxide_solid', '.Lead_sulfate_solid', and so on. The last, very special substance to be included is an electron. This class is called 'Components.ElectronTransfer' and it must be added in order for each electrode to transfer electron from the chemical reaction to the electric circuit (Figure 2C). Each of these substances must be connected to the appropriate solution using a solution port situated in the bottom of the component's icons to indicate that they are all mixed in the solution. By having all these substances, it is possible to implement the chemical reactions. Dragging and dropping the library class 'Components.Reaction' for both chemical reactions, and setting their parameters as an appropriate number of reactants, products and stoichiometry, allows the connection of each substance with the reaction, as expressed in reaction (1) and reaction (2). This setting can be done using the parameter dialog of the cathode chemical reaction (1) as there are four types of substrates (nS=4) with stoichiometric coefficients: one for the first and second reactant, three for the third reactant and two for the fourth reactant (s={1,1,3,2}).

There are also two types of products (nP=2) with stoichiometry: one for $PbSO_4$ and two for water (p={1,2}), following the chemical scheme of the first chemical reaction above. After setting the number of reactants and products, it is possible to connect the substances with reactions. Each instance of reaction has an array of connectors for substrates and an array of connectors for products; the user must be very careful to connect each element of these arrays in the same order as defined by stoichiometric coefficients. This means that, for example, the water must be connected in index 2 to products of the first chemical reaction, because we had already selected the order of products by setting the array of stoichiometric coefficients in reaction (1). The chemical reaction (2) must be set analogically as nS=2, nP=3, p={1,1,2} with connections of substance ports of Pb to substrate[1], HSO_4^- to substrate[2], $PbSO_4$ to product[1], H^+ to product[2] and e^- to product[3], as represented in Figure 2D.

The electrochemical cell has already been implemented at this stage. However, the simulation requires the initial state of substances, which for the fully charged battery means that almost all elements of the cathode are PbO_2 and almost all elements of the anode are Pb. In this state, the sulfuric acid can be very concentrated, which increases the effectiveness of the electrochemical cell. To set this state, it is possible to just double-click on PbO_2 and Pb and set the amount, e.g., 1mol. To set the pure concentrated sulfuric acid we can also set the amount of SO_4^- and H^+ as 1mol. This fully charged ideal state is ready to simulate when it is connected to the electric ground via one of the electric ports of the one electron transfer component.

These batteries can be connected to any electrical circuit that is slowly discharging. For example, if we only connect the simple electric resistance of 1 Ohm as expressed in Figure 2D, then the simulation of the discharging process over 13 hours and 45 minutes gives the results of electric current and electric potential, as can be seen in Figure 3. The exchange of the resistor with a voltage source can simulate the charging process for a discharged cell.

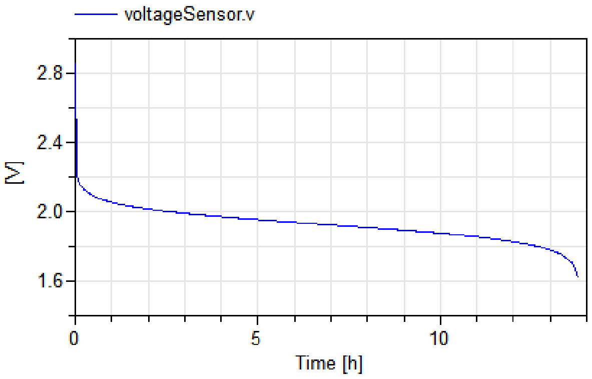


Figure 3. Discharging simulation of the lead-acid battery cell from Figure 2D, with the initial amount of substances as described in the text.

4 Example of the Hydrogen Burning

In contrast with oxidation-reduction reactions, describing processes in lead-acid electrochemical cells, the gaseous reaction of burning hydrogen is very simple:

$$2\ H_2 + O_2 <-> 2\ H_2O \qquad (3)$$

However, this reaction generates a large amount of energy which can be used for mechanical or thermal purposes.

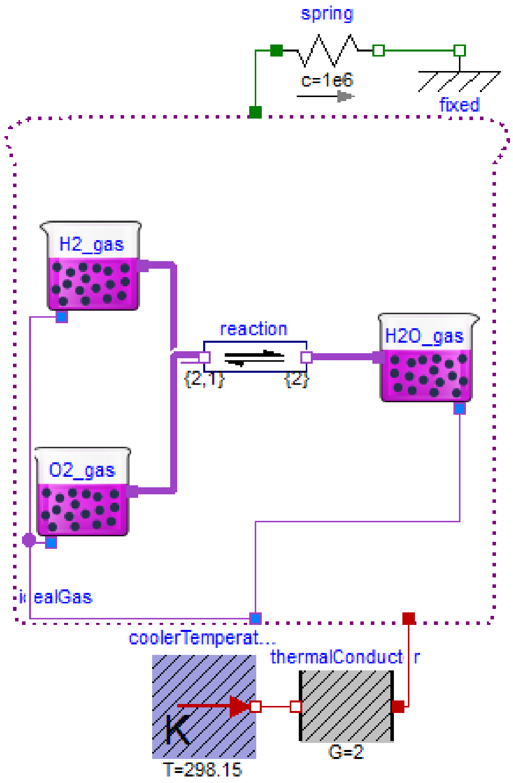


Figure 4. A hydrogen-burning piston with the spring above the piston and cooling to provide an environment with a constant temperature.

Building this model (Figure 4) using the Chemical library is very easy. First, we drag and drop the library class 'Components.IdealGasSolution' into the diagram of our new model, labeled 'idealGas' in Figure 4. For this solution we need to set the area of the piston (e.g., 1 dm^2), where the pressure provides the force of the green mechanical port of the uppermost side. The next parameter is the ambient external pressure surrounding the system (e.g., 1 bar). All three chemical substances of the reaction (1) can be added by dragging and dropping the library class 'Components.Substance'. Because this model uses gases, the state of matter must be changed to some gas, such as the ideal gas prepared as 'Interfaces.IdealGas'. The substance data must be selected to define the appropriate substances such as 'Hydrogen_gas', '.Oxygen_gas' and '.Water_gas' in package 'Examples.Substances'. In addition, the initial amounts of substances can be prepared for the ideal solution of hydrogen and oxygen gases at a ratio 2:1 to attain the chemical equation above, with the expectation that at the end of the burning process, only water vapor would be presented. Therefore, the initial values of H_2 particles could be set to 26 mmol and of O_2 particles as 13 mmol. All substances must be connected with the 'idealGas' using the blue colored solution port situated on the bottom side of each substance and solution. Then, the chemical reaction is inserted into the diagram of this model as library class 'Components.Reaction', and it is set to two substrates (nS=2) with stoichiometry s={2,1} and one product with stoichiometry p={2} to represent the reaction (3). The substances are then connected using violet colored substance connectors with appropriate indexes: H_2 to substrates[1], O_2 to substrates[2] and H_2O to products[1]. At this point, the model is prepared to simulate the conditions of an unconnected heat port and an unconnected mechanical port. This simulation reaches the theoretical ideal of thermally isolated (zero heat flow from/to the solution) and isobaric (zero force generated on piston) conditions.

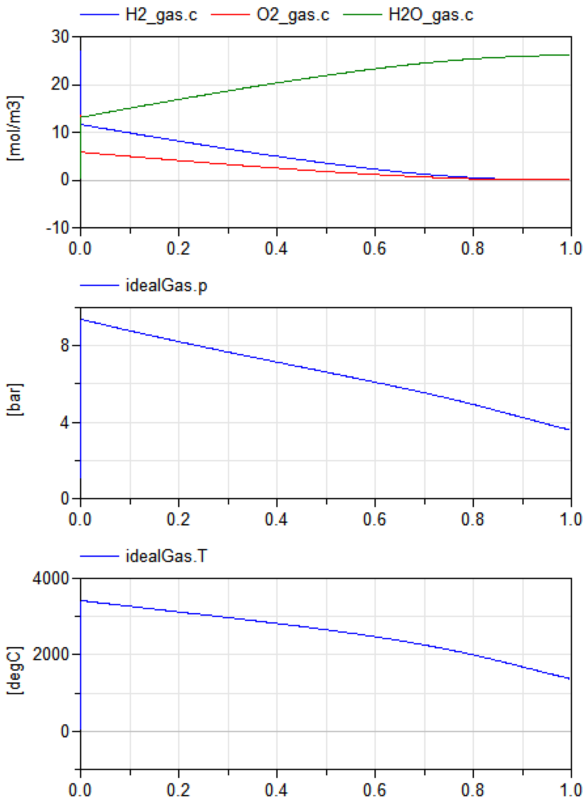


Figure 4. Simulation of the hydrogen-burning experiment in Figure 4. The initial phase of the explosion occurs very rapidly — the temperature reaches immediately 3600°C from 25°C and the pressure reaches 10 bars from 1 bar. This pressure and this temperature are generated because of a very strong spring, which allows the volume to change only by about 8% during the explosion.

However, in the real world, there is always some thermal energy flow from the solution, and this cooling process can be connected using the thermal connector of the Modelica Standard Library 3.2.1. For example, the simple thermal conductor of thermal conductance 2W/K at a constant temperature environment of 25°C is represented in Figure 4. The mechanical power of the engine can be connected to the robust mechanical model. However, in our example we selected only a very strong mechanical spring with a spring constant of 10^6 N/m to stop the motion of the piston in order to generate the pressure. This standard spring component is situated above the solution in Figure 4. The results of this experiment are shown in Figure 5.

5 Example of Chloride Shift

The mature red blood cell (erythrocyte) is the simplest cell in the human body. Its primary function is the transportation of blood gases, such as oxygen O_2 (from the lungs to tissues) and carbon dioxide CO_2 (from tissues to the lungs). The chemical processes behind the gases' transportation are complex because the capacity of water to transport their freely dissolved forms is very low. To transport sufficient amounts of O_2 and CO_2, the gases must be chemically bound to hemoglobin such as described in [10] and/or transported as different substances, which can be present in water in much higher concentrations than their freely dissolved forms allow. Therefore, to transport a sufficient amount of CO_2, it must be changed to HCO_3^- using the chemical reaction:

$$\mathrm{CO2 + H2O <\!\!-\!\!> HCO_3^- + H^+} \tag{4}$$

This reaction takes place mainly inside the red blood cell, because only here it is presented with the enzyme carbonic anhydrase. Therefore, the increase of total carbon dioxide content of blood in tissues and its decrease in lungs are always connected with the chloride shift between blood plasma and the intracellular fluid of erythrocytes, as represented in Figure 6.

The blood plasma and intracellular fluid are divided by the cellular membrane composed of a special, very compact lipid double-layer. A lipophobic compound (not soluble in lipids) without special proteins called membrane channels cannot cross the membrane. Even water molecules must have membrane channels (called aquaporins) in order to cross the cellular membrane. In addition, the chloride shift (also known as the Hamburger shift) is exchanging an aqueous chloride Cl^- for an aqueous bicarbonate HCO_3^- in both directions across the cellular membranes of red blood cells using the membrane channel "Band 3". Each passive membrane channel only allows the equilibration of the electrochemical potentials of the specific permeable ions on both sides of membrane. The different electric potentials on each side of membrane allow their different concentrations to achieve equilibrium.

Conversely, the solution's equilibrium of different ions' compositions on both sides of the membrane creates the measurable electric membrane potential. This process is not so intuitive, because even though neither solution needs to have an electric charge, there can be a non-zero electric potential for permeable ions. This potential for permeable ions at equilibrium is called the Nernst membrane potential and, in the Chemical library, it is a direct mathematical result of the equality of the electrochemical potential of the ion in both solutions.

The intracellular solution must be set at the possible nonzero electric potential (ElectricalGround=false) because, as a result, the membrane potential of the erythrocytes is calculated as -12mV, which agrees with experimental data by [11] in the electrolytes' setting by [12].

In this way, it is possible to model more complex processes of a membrane where chemical reactions of

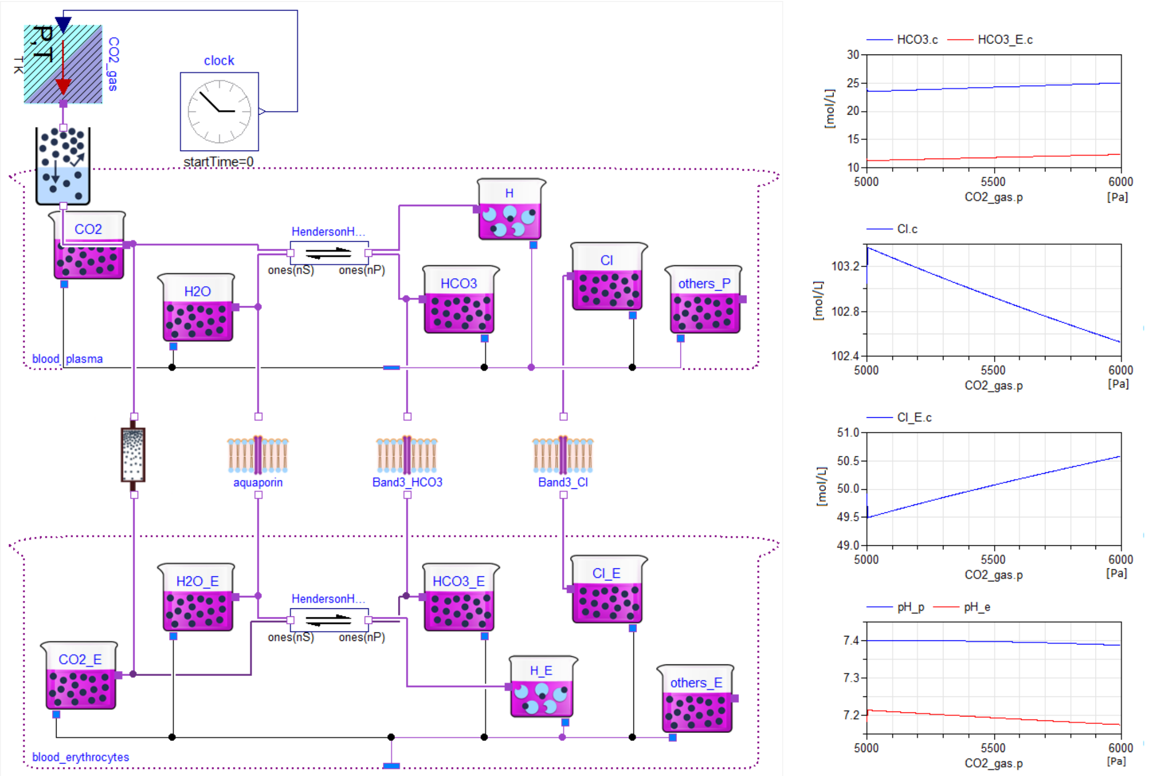


Figure 5. Chloride shift with carbon dioxide hydration with assumption of non-bicarbonate linear acid-base buffering properties of plasma and erythrocytes.

active membrane channels or membrane receptors can both be used.

6 Discussion

Nowadays, alternative free Modelica libraries for chemical calculations exist, such as FCSys v0.2, FuelCellLib 1.0, Modelica_EnergyStorage v3.2.1, BioChem v1.2 or our Physiolibrary v2.3. However, we are not satisfied with these libraries, because none of them are based on equilibrating electrochemical potentials. This lack makes it difficult to establish real equilibria in electrochemical processes, and we believe that it is very difficult to implement any kinetics without realistic equilibria.

This new chemical library is more suited to understanding the detailed electrochemical environment of human cells and cellular electrochemical processes, a task at which the Physiolibrary failed. For example, we found that the equilibrium of osmolarities (as validated and verified for macroscopic and capillary membranes) was not in good agreement with measured data of cellular membranes. The real data of human blood include the total molarity of plasma at 289 mmol/L and the molarity of intracellular space of erythrocytes at 207 mmol/L at osmotic equilibrium, as presented by [12]. These values are definitely not the same, and the explanation for these disproportions can be found in physical chemistry [8]. However, when the electrochemical potential from the original data was calculated, it was found that electrochemical potential is in equilibrium instead of a state of osmolarity. Therefore, equilibrating the electrochemical potential instead of osmolarity can help us to describe each type of membrane and each type of substance, reaching the expected values as measured in osmotic experiments for both organ and cellular membranes.

The library is usable for any chemical or electrochemical process. However, chemical kinetics are not yet seriously validated, so the only assumption is, that the equilibrating time of chemical processes is by orders of magnitude shorter than of other connected domains. Testing has been done through examples in examples package in Dymola 2015.

The mentioned examples, together with many others that have been processed, are implemented and tested in the ‘Example’ package of the library. They are the definition of a very simple and general chemical reaction and also the complex models, such as: the heating of water solutions, an exothermic reaction, the vaporization of water, O_2 and CO_2 gas solubility in aqueous solutions, an enzymatic reaction, a Harned cell (such as the typical pH measurement of an electrochemical cell), water self-ionization, carbon dioxide in a water solution, inorganic phosphate in a water solution, the albumin (blood plasma protein with 218 sides for the binding of H^+) titration model by Figge-Fencl and allosteric models of hemoglobin oxygenation by Monod-Wyman-Changeux. All of these examples illustrate usage of the chemical library’s components, such as the chemical solution, chemical substance and chemical reaction.

We hope, that with reference to the tabulated thermodynamic properties of organic substances, it should be also possible to implement even a complex metabolic, regulations and neural pathways of human physiology using this Chemical library.

7 Acknowledgements

The authors appreciate the partial funding of this work by PRVOUK P/24/LF1, SVV 260157/2015 and FR Cesnet 551/2014.

References

[1] M. Mateják, "Physiology in Modelica," *MEFANET Journal*, vol. 2, pp. 10-14, 2014.

[2] M. Mateják, T. Kulhánek, J. Šilar, P. Privitzer, F. Ježek, and J. Kofránek, "Physiolibrary - Modelica library for Physiology," presented at the 10th International Modelica Conference, Lund, Sweden, 2014.

[3] R. L. Hester, A. J. Brown, L. Husband, R. Iliescu, D. Pruett, R. Summers, *et al.*, "HumMod: a modeling environment for the simulation of integrative human physiology," *Frontiers in Physiology*, vol. 2, 2011.

[4] J. Kofránek, M. Mateják, and P. Privitzer, "HumMod - large scale physiological model in Modelica," in *8th International Modelica Conference*, Dresden, Germany, 2011.

[5] M. Mateják and J. Kofránek, "HumMod–Golem Edition–Rozsáhlý model fyziologických systémů," *Medsoft*, pp. 182-196, 2011.

[6] T. Kulhánek, J. Šilar, M. Mateják, P. Privitzer, J. Kofránek, and M. Tribula, "Distributed computation and parameter estimation in identification of physiological systems," in *VPH conference*, 2010.

[7] F. Casella, M. Otter, K. Proelss, C. Richter, and H. Tummescheit, "The Modelica Fluid and Media library for modeling of incompressible and compressible thermo-fluid pipe networks," in *Proceedings of the Modelica Conference*, 2006, pp. 631-640.

[8] R. G. Mortimer, "Physical Chemistry (Third Edition)," R. G. Mortimer, Ed., ed. Burlington: Academic Press, 2008, pp. 1-1385.

[9] F. G. Donnan, "Theorie der Membrangleichgewichte und Membranpotentiale bei Vorhandensein von nicht dialysierenden Elektrolyten. Ein Beitrag zur physikalisch-chemischen Physiologie," *Zeitschrift für Elektrochemie und angewandte physikalische Chemie,* vol. 17, pp. 572-581, 1911.

[10] M. Mateják, T. Kulhánek, and S. Matoušek, "Adair-based hemoglobin equilibrium with

oxygen, carbon dioxide and hydrogen ion activity," *Scandinavian Journal of Clinical and Laboratory Investigation,* vol. 75, pp. 113-120, 2015/02/17 2015.

[11] M. M. Gedde and W. H. Huestis, "Membrane potential and human erythrocyte shape," *Biophysical journal,* vol. 72, p. 1220, 1997.

[12] J. E. Raftos, B. T. Bulliman, and P. W. Kuchel, "Evaluation of an electrochemical model of erythrocyte pH buffering using 31P nuclear magnetic resonance data," *The Journal of general physiology,* vol. 95, pp. 1183-1204, 1990.