**Rare-Earth Element Electrodialysis Process Modelling**

****

Prepared by Kellie Chong

April 2021

Natural Resources Canada

Canmet Mining

Green Mining Innovation

Water Management Section

**Table of Contents**

[**1.0** **Introduction** 3](#_Toc70610102)

[**2.0** **Background** 3](#_Toc70610103)

[**3.0** **Implementation** 4](#_Toc70610104)

[**3.1.** **Dissociation Equilibrium** 4](#_Toc70610105)

[**3.2.** **Ion exchange equilibrium of membranes** 5](#_Toc70610106)

[**3.3.** **Permeation equations** 6](#_Toc70610107)

[**3.4.** **MATLAB Implementation** 6](#_Toc70610108)

[**3.5.** **Machine learning model** 7](#_Toc70610109)

[**4.0** **Model data validation** 8](#_Toc70610110)

[**5.0** **Conclusions and Future Scope** 14](#_Toc70610111)

[**6.0** **Supporting Information** 15](#_Toc70610112)

[**7.0** **References** 16](#_Toc70610113)

[**8.0** **Appendix A** 17](#_Toc70610114)

1. **Introduction**

Investigation into chelant assisted-Rare Earth Elements (REEs) electrodialysis (ED) for the separation of REEs has been one of the focuses of the Rare Earth Elements and Chromite R&D program at CanmetMining since year 3 (2016-2017). Electrodialysis is a cation- or anion- selective membrane process in which ions of the same magnitude of charge are transported through their respective membranes under an electric potential. These membranes, oriented in parallel in an electrodialysis cell, create an array of dilute and concentrating compartments as illustrated in Figure (1) below. The ability for preferential separation of metal ions in solution by both chelant-assisted and chelant-free ED processes have been successfully demonstrated in previous studies. [1, 2, 3, 4, 5, 6]

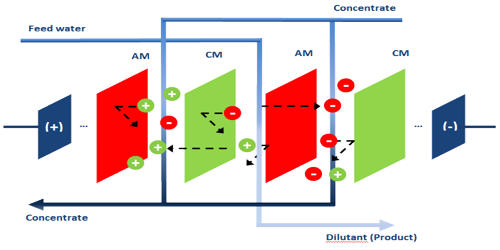


Figure 1: Schematic for a typical electrodialysis cell containing both cationic- and anionic- selective membranes.

Due to the extraordinary circumstances encountered at the end of the 2019-2020 fiscal year as a result of the pandemic, the direction of the REE-ED project shifted from an experimental focus, to a deterministic modelling based-approach for optimizing in-lab experiments. By creating such a model, the aim was to predict the efficiency of REE separation during ED under varying chemical compositions, concentrations, operating parameters, and timespans. Previous mathematical models for electrodialysis have also been achieved in [6, 7, 8, 9, 10]. All of these models applied the principle of electrochemical flux to the Nernst-Planck equation. Herein, we will discuss the recreation of the model presented in Takahashi et al.’s study, which includes considerations for the chemical equilibria of chelating agents in solution and their effect on ion concentrations during ED.

1. **Background**

Positive ions, including protons and REEs in solution will compete in the chelation kinetics. During chelation, a polydentate ligand, such as a REE, will form a stable complex with the chelating agent, and have a characteristic absolute stability constant, KABS. The lower this constant, the more the equilibrium shifts to its ionized components in solution. For REEs in particular, the stability constant for light REEs with various chelating agents used in our experiments tend to be lower than that of heavy REEs. Due to our chelate-REE complexes being too large to diffuse through the membranes, while also bearing a -1 resonant charge, the system can be assumed to be in Donnan equilibrium. This assumption has also been accepted in previous studies, [5, 8] and gives rise to another consideration required for an accurate model – the Donnan potential. Under the Donnan equilibrium, thermal equilibrium, electrical neutrality, and conservation of ions is still maintained, and for this reason, the Nernst-Planck equation still presents an accurate model for the ED system.

1. **Implementation**

The initial approach by proposed by Konstantin in 2019 was to implement a deterministic model based on a method proposed by Takahashi et al. [6] This method is based off the Nernst-Planck equation for electrochemical flux assuming a quasi-steady state electroneutrality. Due to troubles encountered with the simultaneous solving of the first system of non-linear equations, a machine learning approach to a model was also explored. In this section we will describe the architecture and functions used to solve this system. For the mechanistic model, the process described hereafter in sections 3.2 and 3.3 are iterated over the time dimension of the solution, while section 3.1 is only solved for initial solution conditions.

* 1. **Dissociation Equilibrium**

Following basic chemical equilibrium equations and corresponding stability constants (Eq. (1)), the first section aimed to solve for the instantaneous concentrations of the free LREE, HREE, and chelating agent ions in solution. In addition to the concentrations of the cheating agent-bound LREE, and HREE. Depending on the desired conditions of the experiment, the hydronium or sodium ion concentration is calculated in the case of a changing, and constant hydronium ion concentration, respectively.

Now, let us consider a typical electrodialysis experiment with lanthanum, neodymium, and EDTA as the chelating agent as seen in equations (2)-(7) below. Due to the small stability constant of sodium with EDTA (KABS = 43.65), we assume that sodium completely dissociates in solution as reflected in the equations below. Under this assumption, we must consider the stoichiometric ratios of sodium with respect to the chelating agent used. EDTA, HEDTA, and DCTA are di-, tri-, and di- sodium salts, respectively. For this reason, we must include an equivalence term (0.5, 0.33, and 0.5 for EDTA, HEDTA, and DCTA, respectively) when converting our ionized chelating agent concentrations to Na+ concentrations to be further used in the next section.

|  |  |
| --- | --- |
| AB | (1) |
| KABS\_LREE [La3+][EDTA4-] – [(EDTA•La)-] = 0 | (2) |
| [La] - [(EDTA•La)-] - [La3+] = 0 | (3) |
| KABS\_HREE [Nd3+][EDTA4-] – [(EDTA•Nd)-] = 0 | (4) |
| [La] - [(EDTA•La)-] - [La3+] = 0 | (5) |
| [EDTAtot] - [(EDTA•La)-] - [(EDTA•Nd)-] - [EDTAnon-complexing] = 0 | (6) |
| [EDTA4-] – 𝛼4 [EDTAnon-complexing] = 0 | (7) |
| ÷ ( | (7.1) |

To solve equations (2)-(7.1) simultaneously, *fsolve*, a non-linear equation solver function in Python was utilized with parameters x1, x2, x3, x4, x5, x6, and x7 equal to [La-], [EDTA•La], [Nd], [EDTA•Nd], [EDTAnon-complexing], [Na+], and [H+], respectively. Under the condition that [H+] is changing, [Na+] is not solved for and instead assumed to be equal to the total chelating agent concentration divided by the equivalence term. It is important to note that any errors produced from this approximation under the changing hydronium ion concentration condition should not significantly affect the final separation factor as [Na+] is updated at every iteration.

* 1. **Ion exchange equilibrium of membranes**

The second system of equations seen in the model accounts for the ion exchange equilibrium of the membranes. A relationship between the concentration of ions in solution and concentration of ions in the membrane is established as seen in equations (8)-(11) below. The inputs for this function are the concentrations of ionized LREE, HREE, Na, and H computed by either the function outlined in 3.1, or 3.3 on the first, or subsequent iterations, respectively. The solutions attained by simultaneously solving this system with *fsolve* in the *scipy.optimize* library are given by the indices 0, 1, 2, and 3 of an iterable called *memconc* in the model, representing the concentration of the ionized LREE, HREE, Na, and H concentrations in the membrane, respectively. *Q*, seen in equation (11) is the ion-exchange capacity of the membranes, an inherent property of the specific membranes used, and is equal to 1.475 meq/g (1475 mol/m3) in the model. Though the manufacturer no longer produces the membranes used in our experiments, this value was attained as an average from literature. [11]

|  |  |
| --- | --- |
|  | (8) |
|  | (9) |
|  | (10) |
| + + | (11) |
|  |  |

* 1. **Permeation equations**

On the basis that the flux of each ion through the membranes in solution follows the Nernst-Planck equation for electrochemical flux in 3D space, we get equation (12) for the flux of an ion species *i* in solution. The assumption of quasi-steady state and electroneutrality in the membrane given by (11), gives rise to the first order ordinary differential equation seen in (13). Due to the known values of the ion concentrations at t = i-1 from the previous iteration, this problem becomes an initial value problem.

|  |  |
| --- | --- |
|  | (12) |
|  | (13) |

To solve equation (13) for each ion in solution, we get a system of n + 2 non-coupled differential equations, where n is the number of rare earth elements in the solution. In the typical case of our mechanistic model, n = 2. Using the *solve\_ivp* function in the *scipy.integrate* library in python, the algorithm can simultaneously solve these IVPs over ti-1 to ti and return the concentration of the ion in solution at a specified time ti. Different numerical methods available for solving in our model that is to be chosen by the user include *RK45* which implements the algorithm for Runge-Kutta of order 4(5), *DOP853* implementing the explicit Dormand-Prince Runge-Kutta method of 8(5, 3), and lastly *LSODA*, which implements the Adams/BDF method with automatic stiffness detection and switching. The solutions attained from this section on each iteration must be appended to their respective solution arrays, as well as updated as the initial condition to be used for ti+1. Upon termination of the time domain of our entire solution, the separation factor can then be calculated by dividing the final HREE ion concentration by the final LREE ion concentration.

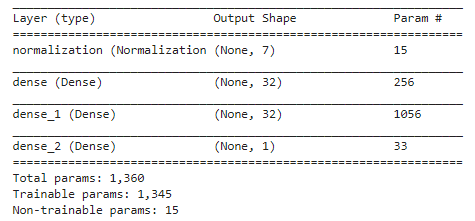
* 1. **MATLAB Implementation**

To assist accommodate for people not as familiar with Python, the same model was reproduced in MATLAB. Currently, this model is applicable for the same set of chelating agents and REEs, with one exception, which is that it currently only models the situation where the hydronium ion concentration is not changing. As a constant hydronium ion concentration is most typical of real in-lab experiments, the model is still relevant. Functions and algorithms used here are analogous to those seen in the Python implementation, and are compartmentalized in a similar fashion for accessible function callback. Rather than importing libraries like you do in Python, the Optimization Toolbox must be installed on from MathWorks in order to use the solvers required for this simulation.

* 1. **Machine learning model**

Alternative to the mechanistic model, a machine learning, or empirical model was also produced in python. Taking advantage of the TensorFlow library, a sequential deep neural network was built with two densely connected hidden layers of 32 + 1 nodes for the data and bias, with linear rectification as seen in Table (1). The training data consisted of 149 samples, comprised of exclusively experiments using La and Eu as the LREE, and HREE, respectively. Six different chelating agents were included in the dataset, including CDTA, DPTA, EDTA, HEDTA, none, and NTA, and one-hot encoding was used to preprocess them. Other parameters used for training this model include the LREE, and HREE concentrations, as well as the chelating agent to HREE ratio, voltage, initial volume, and test duration. In hindsight, since our model only currently comprises of La-Eu experiments, with the majority of them being at 10 V and an initial volume of 1300 mL, perhaps the next iteration of this model could be to clean up this data and remove these for now to improve the accuracy of the model. Here, we ran the model with an 80-20 split for the training, and testing data, respectively. The Adam optimizer was used for training, with a learning rate of 0.01, and batch sizes of 20 over 300 epochs. The loss function for the training, and validation sets calculating the mean squared error over epochs is illustrated in Fig. 2. Successfully trained models can then be saved as a JSON file with its corresponding weights calculated from the training and loaded to future projects.

Table 1: Deep neural network architecture used for the machine learning model.



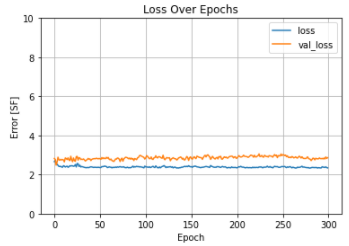


Figure 2: A plot of loss over epochs generated from training the model with the architecture outlined in Table 1.

1. **Model data validation**

As with any mathematical model, data validation is an important step to ensure that the results computed by the model are consistent experimental results. One potential source of error in our data validation process is the lack of duplicate experimental results available for data validation on the experimental end. For this reason, it is without certainty that our computed results accurately represent realistic concentrations and thus, separation factors. As an intermediate, we will accept the experimental values to be free of error and compare them to our model results. In particular, four random experiments containing La and Nd as the LREE and HREE, respectively, were used for the data validation process as outlined in Table 1 below. The computed separation factors are displayed on the inset of the graphs in Figures (4)-(6). Corresponding percent errors range from 3.08 to 77.90%. Other LREE-HREE combinations are not available for evaluation using the deterministic model as stability, membrane selectivity, and self-diffusivity constants for elements used in in-lab experiments are not currently available.

Table 2: Outline of operating parameters in the experiments used for data validation in the mechanistic Python model.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Batch Number** | **Concentration** | **Chelating Agent** | **CA:HREE** | **Time (min)** | **SF** |
| 23110 | 825-825 | EDTA | 2:1 | 240 | 2.70 |
| 23127 | 825-825 | HEDTA | 2:1 | 300 | 4.70 |
| 23990 | 825-1030 | HEDTA | 2:1 | 300 | 2.24 |
| 23152 | 825-825 | CDTA | 1:1 | 300 | 2.72 |

B

A

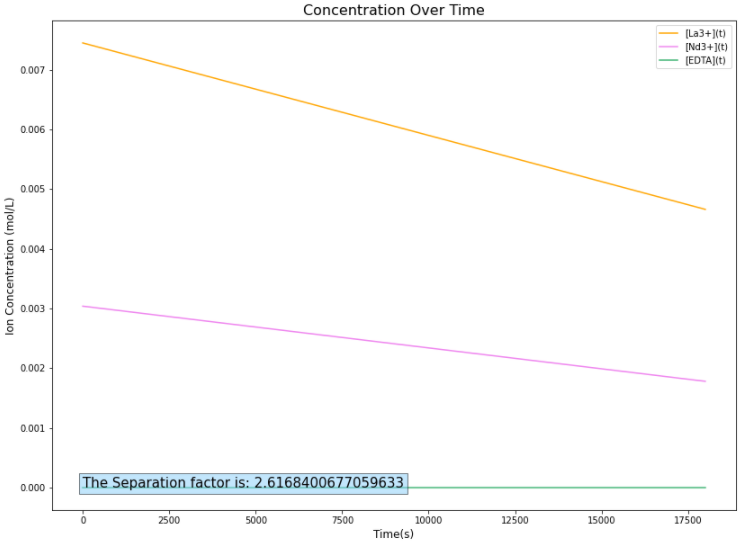
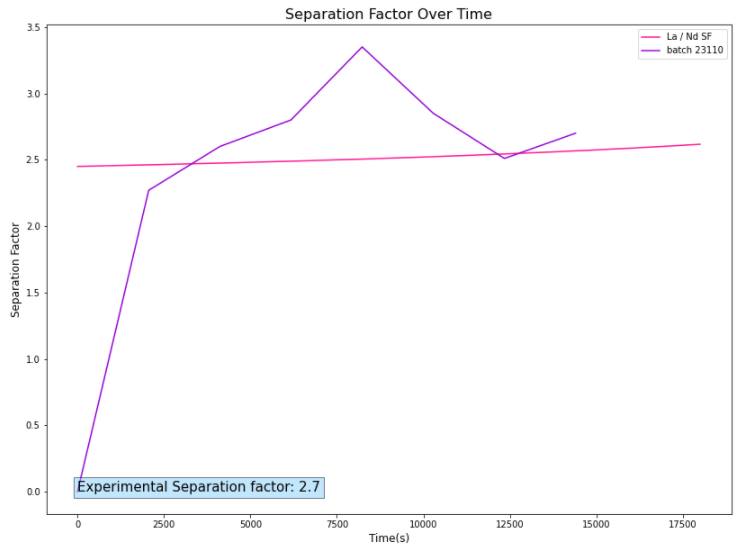
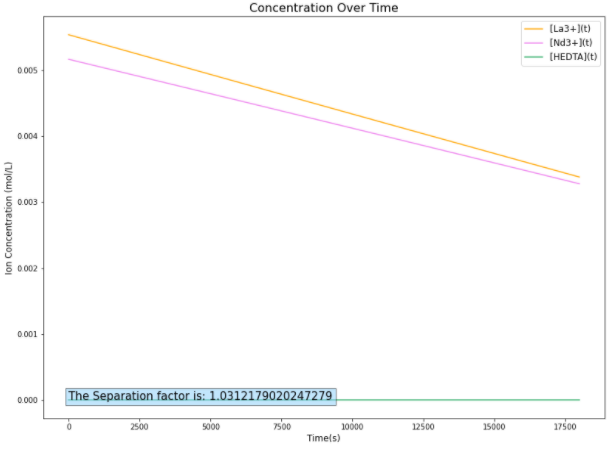
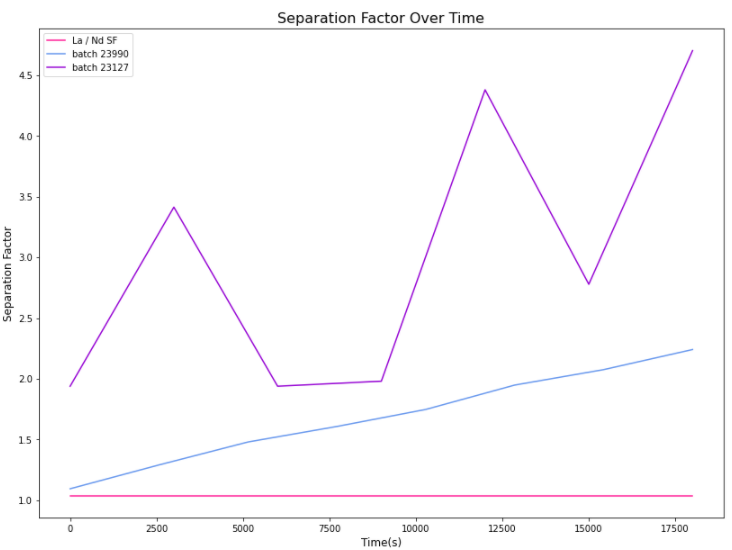
 

Figure 3: (A) Model performance with EDTA, 825 ppm La, 825 ppm Nd, CA:HREE = 2:1, Q = 1.475 meq/g, pH = 5, and t = 1800 s. The calculated separation factor at tfinal = 2.62. (B) Comparison of the separation factor from batch number 23110, and the model (3.08% error).

B

A

Figure 4: (A) Model performance with HEDTA, 825 ppm La, 825 ppm Nd, CA:HREE = 2:1, Q = 1.475 meq/g, pH = 5, and t = 1800 s. The calculated separation factor at tfinal = 2.62. (B) Comparison of the separation factor from batch 23127 (825 ppm-825 ppm), and batch 23990 (825 ppm-1030 ppm), and the model.

A

B

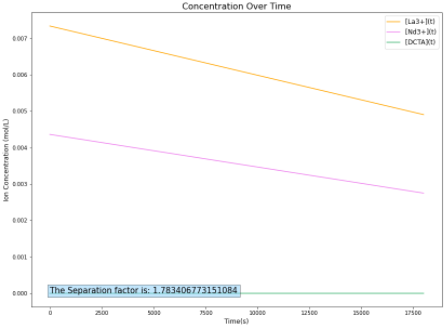
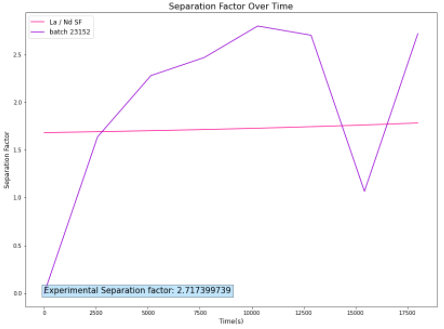
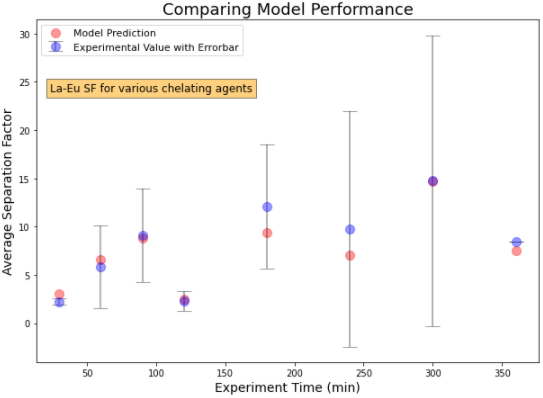
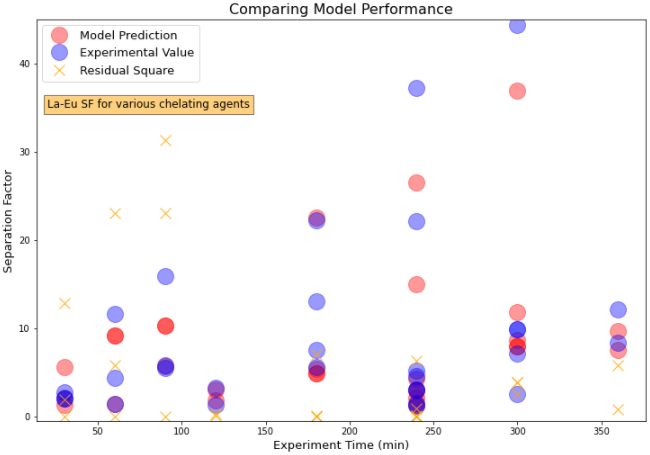
 

Figure 5: (A) Model performance with CDTA, 825 ppm La, 825 ppm Nd, CA:HREE = 1:1, Q = 1.475 meq/g, pH = 4, and t = 1800 s. The calculated separation factor at tfinal = 1.78. (B) Comparison of the separation factor from batch 23152, and the model (34.1% error).

Conversly, the machine learning model was trained off La-Eu experiments and their corresponding separation factors yielded an average loss of 2.7 separation factor units. Considering the wide spread of separation factors present in the training data (~45 SF units), an average loss of 2.7 is not significant. Taking 30 random data points consisting of various chelating agents, concentrations, and operating conditions, yields the graph illustrated in Figure (7A). Looking at this graph, it is clear that the majority of residual square values lie close to zero, and that greater residual square values are seen at the beginning of the experiment. This observation is consistent with the error propogating from small values of removal percentages, enabling small deviations to largely impact the separation factors at the beginning of experiments. Cleaning up the data a bit by taking the average and standard deviation at each time point, we get Figure (7B). The large spread of separation factors is due to the presence of experiments with differing operating parameters, giving rise to a wide range of separation factors.



B

A

Figure 6: (A) Validation data performance of the machine learning model. Here, n = 30 data points taken from the training set. Varying amounts of data points were selected from each time point. Residual square values are also plotted on the same graph and denoted by a yellow ‘X’. (B) Cleaned up plot of 6(A). The bar here represents the standard deviation of the experimental values at each time point.

Furthermore, batch 23056 was also used for model validation using the machine learning model. Operating parameters for this test are as follows: chelating agent: EDTA, La: 825 ppm, Eu: 825 ppm, 1:1 (HREE:CA ratio), 10 V, and initial volume: 1300 mL. Due to the small number of experimental results with this exact set of parameters (3) seen in the training data, a large error is expected. It is suggested to increase the number of points in training dataset (n > 500), while simultaneously reducing the number of less important parameters for training a more accurate model.

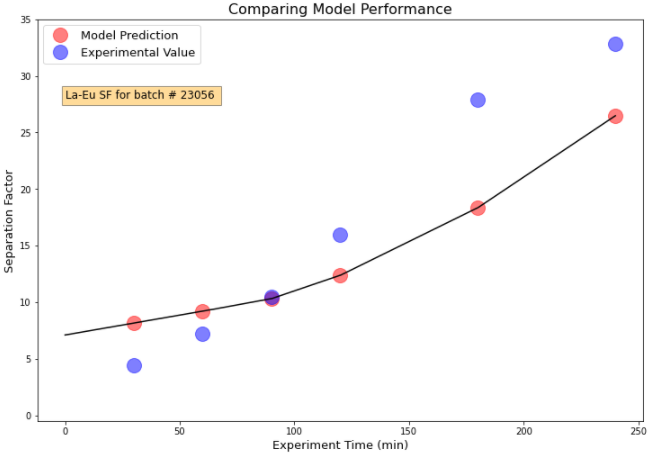


Figure 7: Comparison of the machine learning model values with the experimental results for batch number 23506. Operating parameters were 825 ppm La, 825 ppm Eu, EDTA in a 1:1 ratio with Nd, V = 10 V, and initial volume = 1300 mL.

Surprisingly, although the MATLAB implementation employed all the same functions and algorithms, the results seen here are slightly different from that of the Python implementation. Here, the solver behaved as expected when the chelating agent to HREE ratio was 1:1, however, when the ratio increased to 2:1, the solution for the concentrations over time became negative, producing inaccurate separation factors. The percent errors in this model reached as low as 5.2% for a solution containing CDTA, at concentrations of 825 ppm for both the LREE and HREE, and 1:1 ratio for the CA:HREE at t = 240 min, meanwhile SFs for the 2:1 CA:HREE test cases yielded SFs up to ~4-fold higher than experimental values as illustrated in Figures (8)-(12).

B

A

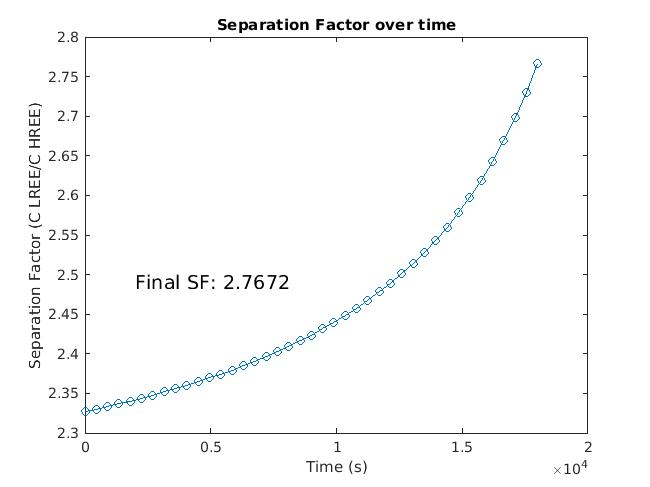
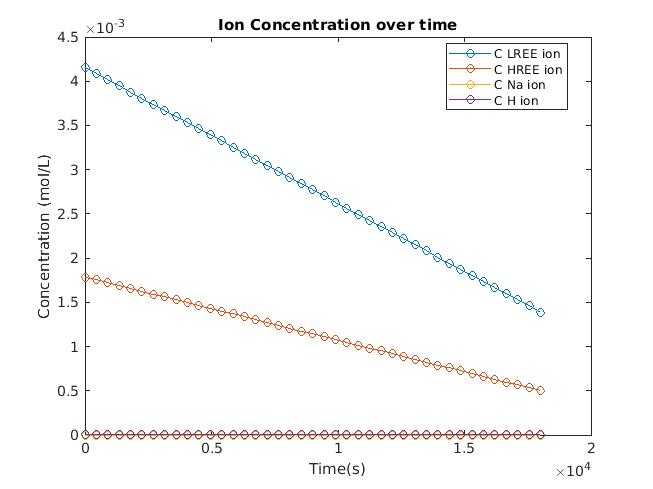


Figure 8: MATLAB implementation of the model under the same operating conditions as batch number 23152 (5.2% error at t = 240 min). (A) Concentration over time plot. (B) Separation factor over time plot.

B

A

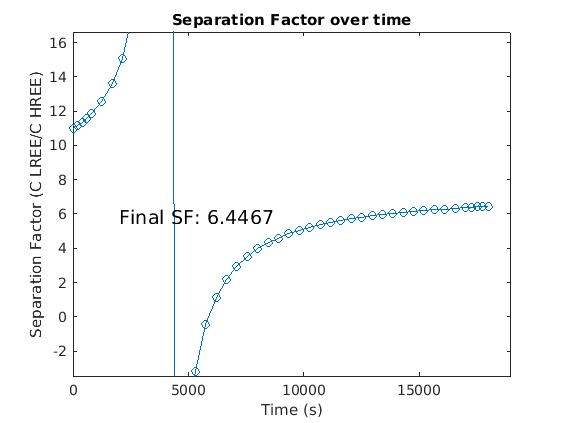
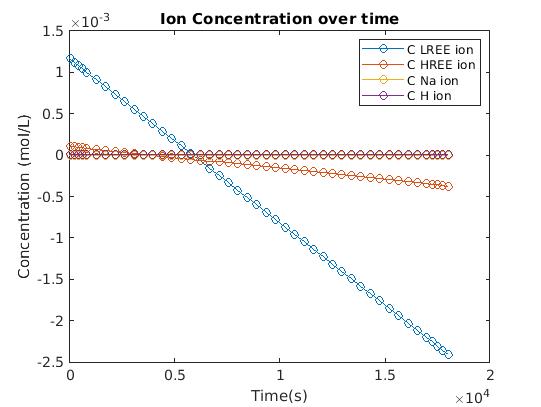
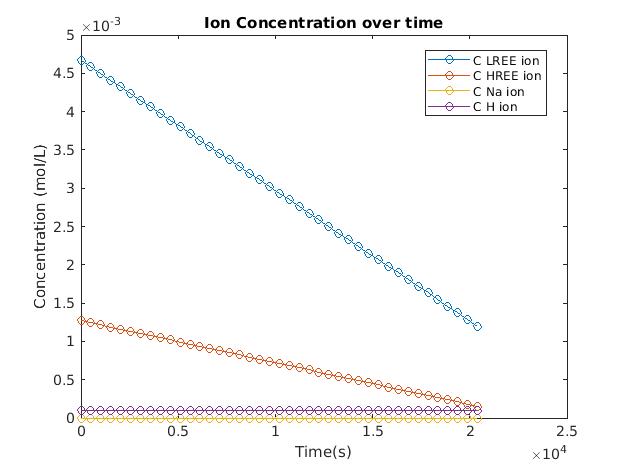


Figure 9: MATLAB implementation of the model under the same operating conditions as batch number 23110 (CA: EDTA, 825 ppm La, 825 ppm Nd, pH = 5, CA:HREE = 2:1). (A) Concentration over time plot. (B) Separation factor over time plot. The fluctuating behaviour in SF can be attributed to the shift from a positive to negative concentration of ionized LREE.



B

A

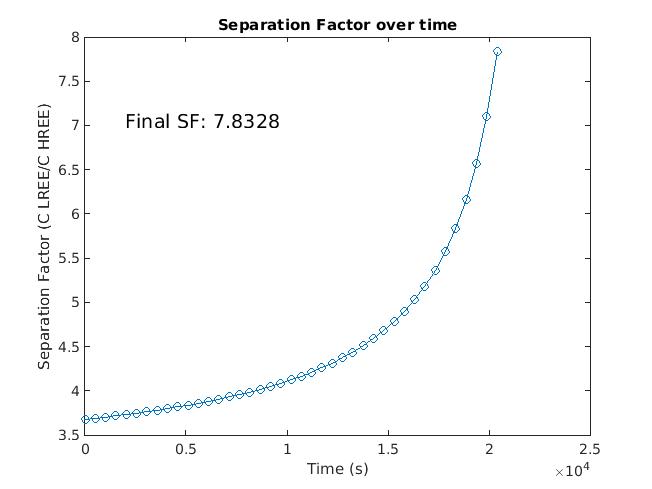


Figure 10: MATLAB implementation of the model under the following operating conditions: CA: EDTA, 825 ppm La, 825 ppm Nd, pH = 5, CA:HREE = 1:1. Under the same conditions, a separation factor of 6.1 was achieved in an in-lab experiment. (A) Concentration over time plot. (B) Separation factor over time plot.

B

A

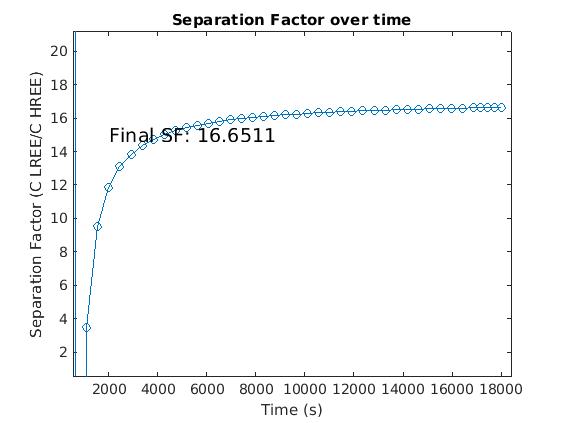
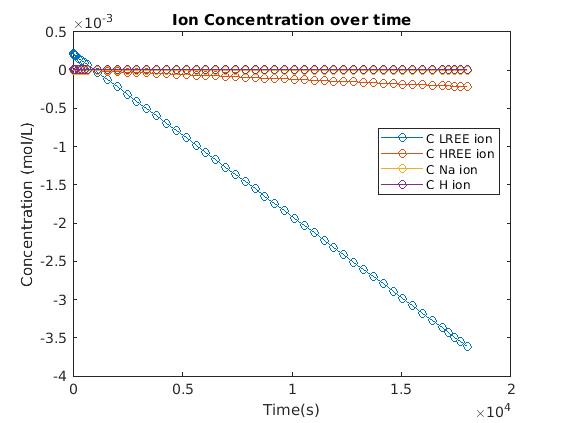


Figure 11: MATLAB implementation of the model under the same operating conditions as batch number 23127 (CA: HEDTA, 825 ppm La, 825 ppm Nd, pH = 5, CA:HREE = 2:1). (A) Concentration over time plot. (B) Plot of the separation factor over time graph, shifted to the right.

Interestingly, attempting to model the conditions present in batch number 23990 where the concentration of Nd was 1030 ppm, and the ratio of CA:HREE was 2:1 resulted in a flux solution with complex numbers. This can be attributed to negative concentrations being returned by the first system of nonlinear equations described in 3.1. As a result, complex solutions are generated in the subsequent solutions (3.2 and 3.3). It was observed that whenever the reason for termination of the first system of equations (3.1) was due to the relative size of the current step being less than the value of the step size tolerance squared, one or more negative concentration results would result. This behaviour was most prominent under operating conditions where the chelating agent concentration was much greater than the REE concentrations. Figure 12 illustrates the stability of a 1:1 CA:HREE composition, with the rest of the conditions being the same as batch 23990.

A

B

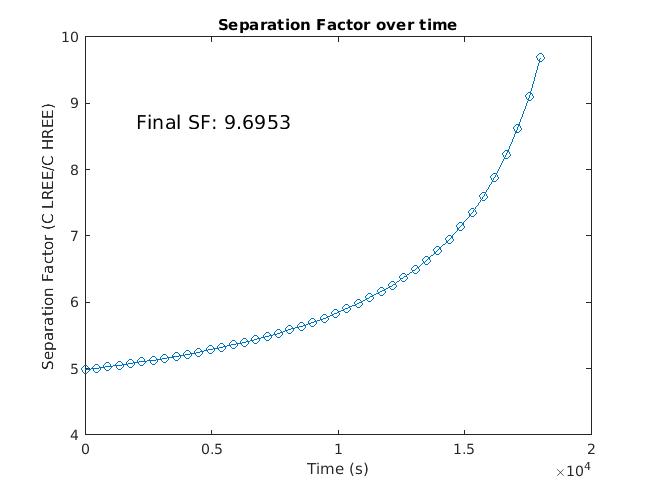
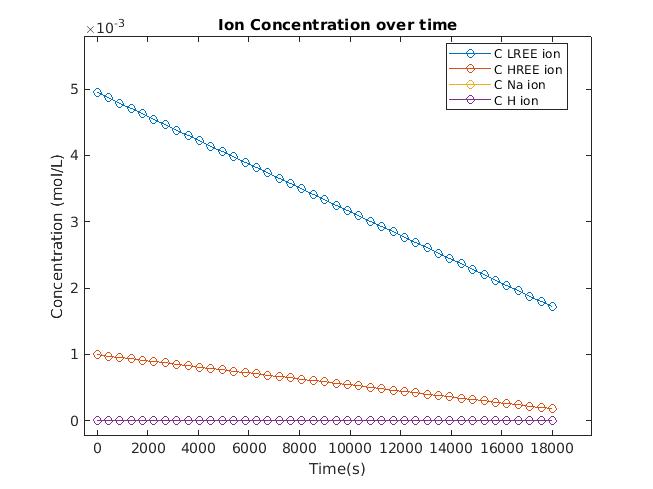


Figure 12: MATLAB implementation of the model under the following operating conditions: CA: HEDTA, 825 ppm La, 1030 ppm Nd, pH = 5, and CA:HREE = 1:1. (A) Concentration over time plot. (B) Separation factor over time plot with the final separation factor after 300 min being 9.6953.

1. **Conclusions and Future Scope**

The simplified Nernst-Planck equation for electrochemical flux, coupled with basic ion-exchange equilibria thermodynamics was employed as three complimentary steps for predicting the separation factor of REEs in solution during the electrodialysis process. We then presented the process, and functions used for developing the code for the representative Takahashi et al. model using both Python, and as an alternative, MATLAB. Comparison between the methods proved that the MATLAB implementation, suggested a more accurate final outcome at 1:1 CA:HREE ratios and constant pH. However, for versatility, the Python implementation is able to better accommodate dynamic operating parameters. While both these deterministic models lacked consistency in accuracy, the deep neural network machine learning model was able to overcome this. However, the one caveat to the machine learning model is that the prediction of the separation factor at the beginning of the experiment is less accurate due to large deviations in the training data for smaller values of t. Similarly, due to lab constraints and lack of duplicated experiments, there is not currently enough data to create a highly accurate model.

Though we know that both concentration and voltage do not have linear relationships with the separation factor, Takahashi et al.’s deterministic model made many assumptions to overcome this. In contrast to other electrodialysis models seen in literature, Takahashi et al.’s model does not consider some important parameters in calculating the electrochemical flux. These parameters include but are not limited to the flow rate, thickness of the dilute compartment, channel length, and resistance of the dilute and concentrating solutions. [7] For this reason, some suggestions for future iterations include:

1. Empirically confirming parameters such as Q, and I, used in the Takahashi et al. model
2. Make the model more applicable to real in-lab experiments by adding constants for more frequently used REEs such as Europium
3. Include considerations for the number of membranes in the stack, flow rate, dimensions of the ED cell, ion mobility (Nernst-Einstein equation), and Donnan potential created by complexed agents in the dilute compartment (See [7, 8])
4. Validating experimental results by running duplicate experiments, this will in turn also improve the machine learning model’s dataset
5. Cleaning up the data in the machine learning model by removing less important parameters, and increasing the size of the dataset (See [9, 12])
6. **Supporting Information**

Link to GitHub repository: <https://github.com/KellieChong/REE-ED-Model>

Hover over text to click on embedded links below:

[Link to Python Implementation](https://colab.research.google.com/drive/1ipfhbnEAqpKv9z1Q57CMPZvYqZ_P3dnu?usp=sharing)

[Link to MATLAB Implementation](https://drive.matlab.com/sharing/77909ef4-03d8-4da1-89f0-a715b564aa3f)

[Link to machine learning model](https://colab.research.google.com/drive/1Nk2zFW963xWj-BWfKJG3X-sJZ0T1nEkb?usp=sharing)

1. **References**

|  |  |
| --- | --- |
| [1] | T. Sata, "Studies on ion exchange membranes with permselectivity for specific ions in electrodialysis," *Joumal of Membrane Science,* vol. 93, no. 2, pp. 117-135, 1994. |
| [2] | T. Sata, T. Sata and W. Yang, "Studies on cation-exchange membranes having permselectivity between cations in electrodialysis," *Journal of Membrane Science,* vol. 206, no. 1-2, pp. 31-60, 2002. |
| [3] | H. Ohya, A. Hakamaya and Y. Hiraoka, "Theoretical studies on the separation of different ionic species of the same charge sign using cascaded electrodialysis with reflux," *Desalination,* vol. 94, no. 2, pp. 167-199, 1993. |
| [4] | A. J. Chaudhary, J. D. Donaldson, S. M. Grimes and N. G. Yasri, "Separation of nickel from cobalt using electrodialysis in the presence of EDTA," *Journal of Applied Electrochemistry,* vol. 30, no. 4, pp. 439-445, 2000. |
| [5] | A. T. Cherif, A. Elmidaoui and C. Gavach, "Separation of Ag+, Zn2+ and Cu2+ ions by electrodialysis with a monovalent cation specific membrane and EDTA," *Journal of Membrane Science,* vol. 96, no. 1, pp. 39-49, 1993. |
| [6] | H. Takahashi, K. Miwa and K.-i. Kikuchi, "Group Separation of Rare-Earth Elements by Use of Electrodialysis with a Complexing Agent," *Journal of Ion Exchange,* vol. 4, no. 3, pp. 183-193, 1993. |
| [7] | M. Sadrzadeh, A. Kaviani and T. Mohammadi, "Mathematical modeling of desalination by electrodialysis," *Desalination,* vol. 206, no. 1-3, pp. 538-546, 2007. |
| [8] | H. Zhu, B. Yang and Y. Wu, "Ion transfer modeling based on Nernst–Planck theory for saline water desalination during electrodialysis process," *Asia-Pacific Journal of Chemical Engineering,* vol. 15, no. 2, pp. 2410-2421, 2020. |
| [9] | M. Sadrzadeha, T. Mohammadia, J. Ivakpourb and N. Kasiri, "Separation of lead ions from wastewater using electrodialysis: Comparing mathematical and neural network modeling," *Chemical Engineering Journal,* vol. 144, no. 3, pp. 431-441, 2008. |
| [10] | Z. Zourmand, F. Faridirad, N. Kasiri and T. Mohammadi, "Mass transfer modeling of desalination through an electrodialysis cell," *Desalination,* vol. 359, pp. 41-51, 2015. |
| [11] | M. Ahmad, A. A. Qaiser, N. U. Huda and A. Saeed, "Heterogeneous ion exchange membranes based on thermoplastic polyurethane (TPU): effect of PSS/DVB resin on morphology and electrodialysis," *RSC Advances,* vol. 10, no. 6, pp. 3029-3039, 2020. |
| [12] | G. Jing, W. Du and Y. Guo, "Studies on prediction of separation percent in electrodialysis process via BP neural networks and improved BP algorithms," *Desalination,* vol. 291, pp. 78-93, 2012. |

1. **Appendix A**

Table 3: Table comparing different membranes and their properties taken from [11]. IEC is called Q in our model.

| **Sr. no.** | **Name** | **Binder : resin/ filler ratio** | **Type** | **tmi (Na+)** | **IEC (meq. g−1)** | **Permselectivity** | **Water uptake (%)** | **Ref.** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | PC | 40 : 60 | Heterogeneous | 0.91 | 1.8 | 0.86 | 27 | [40](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit40) |
| 2 | PVC/ABS | 50 : 50 | Heterogeneous | 0.77 | 1.6 | 0.65 | 25.8 | [41](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit41) |
| 3 | SBR/PC | 50 : 50 | Heterogeneous | 0.92 | 1.8 | 0.88 | 27.8 | [42](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit42) |
| 4 | ABS/HIPS | 50 : 50 | Heterogeneous | 0.88 | 1.5 | 0.80 | 29.5 | [43](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit43) |
| 5 | PVC | 30 : 70 | Heterogeneous | 0.92 | 3.0 | — | — | [10](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit10) |
| 6 | PVC | 50 : 50 | Heterogeneous | 0.90 | — | 0.84 | 18 | [44](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit44) |
| 7 | S-PVC | 40 : 60 | Heterogeneous | 0.90 | 1.9 | 0.84 | 40 | [40](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit40) |
| 8 | CMI 7000 | — | Heterogeneous | 0.94 | — | — | — | [45](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit45) |
| 9 | PVC–ZrAIP | 25 : 75 | Heterogeneous | 0.89 | — | 0.82 | 10.3 | [47](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit47) |
| 10 | CMF | — | Homogeneous | 0.95 | — | — |  | [45](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit45) |
| 11 | FKS | — | Homogeneous | 0.96 | — | — | — | [45](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit45) |
| 12 | SPEEK | — | Homogeneous | 0.92 | 1.1 | 0.86 | 18.5 | [46](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit46) |
| 13 | CMV (Asahi) | — | Homogeneous | — | 2.4 | 0.95 | 25 | [39](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit39) |
| 14 | HJC (Asahi) | — | Heterogeneous | — | 1.8 | — | 51 | [39](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit39) |
| 15 | CMX (Neosepta) | — | Heterogeneous | — | 1.5–1.8 | 0.97 | 25–30 | [39](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit39) |
| 16 | CMS (Neosepta) | — | Heterogeneous | — | 2.0 | — | 38 | [39](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit39) |
| 17 | Nafion 117 (DuPont) | — | Homogeneous | — | 0.9 | 0.97 | 16 | [39](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit39) |
| 18 | CMD | — | Homogeneous | 0.94 | — | — |  | [45](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit45) |
| 19 | FKB | — | Homogeneous | — | 1.2 | — | 30 | [45](https://pubs.rsc.org/en/content/articlelanding/2020/ra/c9ra06178a#cit45) |
| 20 | TPU-PSS/DVB | 60 : 40 | Heterogeneous | 0.87 | 1.08 | 0.80 | 45 | The present research |
| 50 : 50 | 0.95 | 1.2 | 0.92 | 51 |
| 45 : 55 | 0.89 | 1.3 | 0.82 | 54 |
| 40 : 60 | 0.84 | 1.4 | 0.73 | 63 |