



HAL
open science

Simulateur de voltampérométrie cyclique sous Python dédié à un micro-capteur à base de graphène

Aïda Bensebaa, Yves Pain, Périne Landois, Sylvie Contreras, Christine Vautrin-UI, Jimmy Nicolle, Vincent Astié, Jean-Manuel Decams, Christophe Coillot

► **To cite this version:**

Aïda Bensebaa, Yves Pain, Périne Landois, Sylvie Contreras, Christine Vautrin-UI, et al.. Simulateur de voltampérométrie cyclique sous Python dédié à un micro-capteur à base de graphène. Les Journées Electrochimiques 2024, Jul 2024, Saint - Malo, France. hal-04701580

HAL Id: hal-04701580

<https://hal.science/hal-04701580v1>

Submitted on 18 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Simulateur de voltampérométrie cyclique sous Python dédié à un micro-capteur à base de graphène.

Aïda Bensebaa,^a Yves Pain,^b Périne Landois,^a Sylvie Contreras,^a Christine Vautrin-UI,^b Jimmy Nicolle,^b Vincent Astié,^c Jean-Manuel Decams^c, Christophe Coillot^a

^a L2C Laboratoire Charles Coulomb, UMR5221 - Université de Montpellier, France

^b ICMN Interfaces, Confinement, Matériaux et Nanostructures, UMR7374 - Université d'Orléans, France

^c Annealsys, 139 rue des Walkyries, 34000 Montpellier, France

Contexte - Projet ANR MIGRASENS

- Développement de micro-capteurs électrochimiques à base de graphène.
- Détection sensible et sélective de micropolluants comme l'isoproturon[1][2].
- Conformité avec les directives européennes sur les contaminants prioritaires (pesticides, résidus médicamenteux ...).

Objectifs

- Développement d'un code Python pour simuler l'expérience de voltamétrie cyclique (3 électrodes) indépendamment d'un logiciel de contrôle et d'analyse électrochimique type NOVA. Cela permet; une personnalisation une flexibilité dans la modélisation, une adaptation avec possibilité d'extension, une meilleure compréhension, une reproductibilité des expériences, enfin un partage libre et gratuit.
- Utilisation des schémas numériques pour résoudre les équations différentielles ordinaires (ODE) ou partielles (EDP): Euler explicite et implicite
- Besoin de modéliser le comportement des processus électrochimiques étudiés.

Équations But : modéliser la dynamique des concentrations et des courants dans une cellule électrochimique.

1^{ère} Loi de Fick

$$J = -D \frac{\partial C}{\partial x}$$

2^{ème} Loi de Fick

$$\frac{\partial C}{\partial t} = D \frac{\partial^2 C}{\partial x^2}$$

Butler-Volmer

$$J = k_f C_{Ox} - k_b C_{Red}$$

$$k_f = k_0 e^{-\alpha \frac{nF}{RT} (E - E^0)}$$

$$k_b = k_0 e^{(1-\alpha) \frac{nF}{RT} (E - E^0)}$$

Nernst

$$E = E_0 + \frac{RT}{nF} \ln \frac{C_{Ox}}{C_{Red}}$$

Modèle Numérique

Méthode EULER EXPLICITE

VS

EULER IMPLICITE

```
for j in range(0,Nx):
    COx[0,j] = C0
    CRed[0,j] = 0
for i in range(0,Nt-1):
    COx[i,Nx-1] = C0
    CRed[i,Nx-1] = 0
```

Conditions initiales identiques aux deux méthodes

Discretisation spatiale et temporelle en utilisant la méthode des différences finies à la 2^{ème} loi de Fick

```
COx[i,j] = COx[i-1,j] + maille * (COx[i-1,j-1] - 2*COx[i-1,j] + COx[i-1,j+1])
CRed[i,j] = CRed[i-1,j] + maille * (CRed[i-1,j-1] - 2*CRed[i-1,j] + CRed[i-1,j+1])
CRed[i,j] = CRed[i-1,j] + maille * (CRed[i-1,j-1] - 2*CRed[i-1,j] + CRed[i-1,j+1]) - k1*dt*CRed[i-1,j-1]
CRed[i,j] = CRed[i-1,j] + maille * (CRed[i-1,j-1] - 2*CRed[i-1,j] + CRed[i-1,j+1]) - 2*k2*dt*math.pow(CRed[i-1,j-1],2)
```

```
JOx[i] = (kf[i]*COx[i,1] - kb[i]*CRed[i,1]) / (1 - (kf[i] + kb[i]) * dx/D)
JRed[i] = -JOx[i]
```

Calcul des flux de courants

```
COx[i,0] = COx[i,1] + JOx[i]*dx/D
CRed[i,0] = CRed[i,1] + JRed[i]*dx/D
```

COxbs[i,Nx-1] = C0
CRedbs[i,Nx-1] = 0
Conditions Limites (CL)

```
I[i] = -n*F*S*JOx[i]
```

```
for j in range(1,Nx-1):
    A[j,j] = 1+2*maille
    A[j,j-1] = - maille
    A[j,j+1] = - maille
A[0,0] = 1+2*maille
A[0,1] = - maille
A[Nx-1,Nx-1] = 1+2*maille
A[Nx-1,Nx-2] = - maille
Ainv = np.linalg.inv(A)
```

Passage par la forme matricielle nécessaire

```
for i in range(1,Nt):
    for j in range(0,Nx-1):
        COxvec[j] = COxbs[i-1,j]
        CRedvec[j] = CRedbs[i-1,j]
```

```
JOxbs[i-1] = (kf[i-1]*COxbs[i-1,1] - kb[i-1]*CRedbs[i-1,1]) / (1 - (kf[i-1] + kb[i-1]) * dx/D)
JRedbs[i-1] = -JOxbs[i-1]
```

```
COxvec[0] = (1+maille)*COxbs[i-1,1] + maille*dx/D*JOxbs[i-1]
CRedvec[0] = (1+maille)*CRedbs[i-1,1] + maille*dx/D*JRedbs[i-1]
COxvec[Nx-1] = COxbs[i-1,Nx-1] + maille*C0
CRedvec[Nx-1] = CRedbs[i-1,Nx-1] + maille*0
```

CL à la surface de l'électrode

CL loin de l'électrode

```
Ibs[i-1] = -n*F*S*JOxbs[i-1]
```

```
COxvec = np.dot(Ainv,COxvec)
CRedvec = np.dot(Ainv,CRedvec)
```

Calcul du vecteur de C_{Ox} et C_{Red} au pas de temps suivant

```
for j in range(0,Nx):
    COxbs[i,j] = COxvec[j]
    CRedbs[i,j] = CRedvec[j]
```

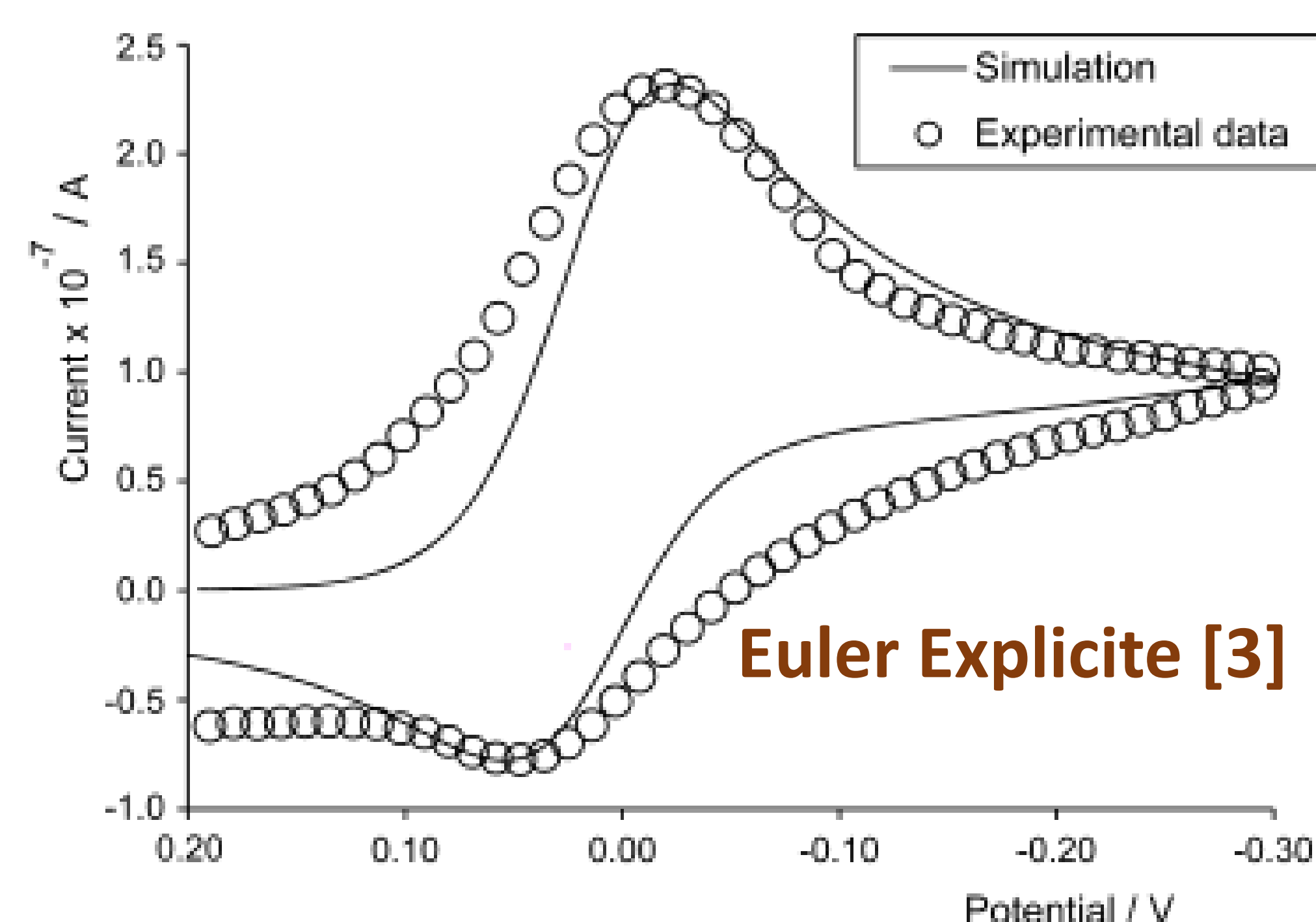
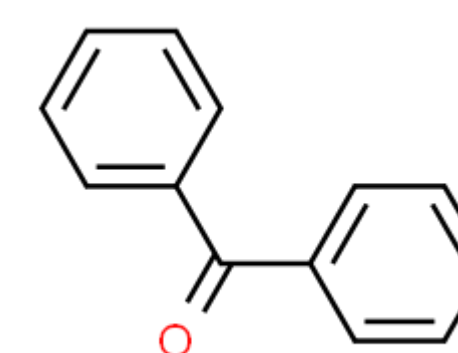
Remplissage des matrices

```
COxbs[i,Nx-1] = C0
CRedbs[i,Nx-1] = 0
```

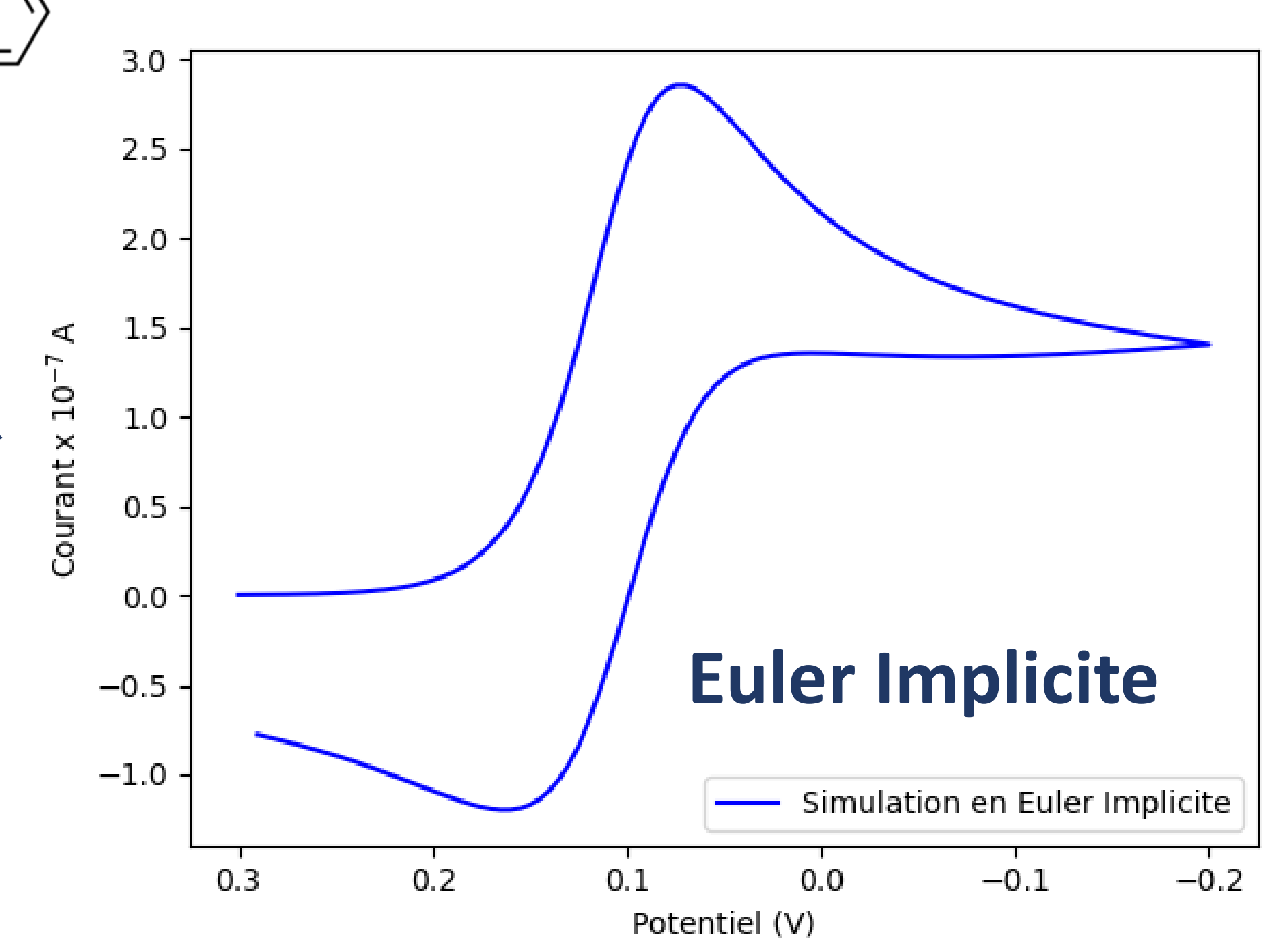
CL

Simulations numériques

La molécule cible choisie ici est le Benzophénone, inspiré du travail de J.H. Brown[3]



Paramètres de Jay H. Brown :
 $C_{Ox,Bulk} = 6.1 \times 10^{-8} \text{ mol/cm}^3$,
 $E^0 = 0.013 \text{ V}$,
 $n = 1$,
 $D = 1 \times 10^{-5} \text{ cm}^2/\text{s}$,
 $A = 2.54 \times 10^{-2} \text{ cm}^2$,
 $v = 0.04 \text{ V/s}$,
 $k^0 = 1 \text{ cm/s}$,
 and $k_2 = 1.6 \times 10^6 \text{ cm}^3/\text{mol s}$.



Conclusions et Perspectives

Avancées

- Si il existe un modèle numérique de voltamétrie cyclique utilisant le schéma d'Euler Explicite, ce dernier est limité et est conditionnellement stable, c'est-à-dire que pour assurer la stabilité de la solution, alors le pas de temps doit être suffisamment petit en fonction du pas spatial. Si ces conditions ne sont pas effectives, alors la solution peut produire des résultats erronés. C'est pourquoi nous développons un modèle sur Euler Implicite, permettant des simulations inconditionnellement stable, versatiles, ouvert et polyvalent dans des conditions extrêmes.
- On obtient des simulations plus précises et plus robustes, car moins d'accumulation d'erreur et réduction du nombre total de pas de temps.
- On peut commencer à comparer les mesures expérimentales entre Euler Explicite et Euler Implicite.

Perspectives Futures

- Corriger les éventuelles erreurs sur le modèle d'Euler Implicite, notamment sur les conditions limites.
- Développement du simulateur pour incorporer diverses conditions expérimentales, comme les onde carrée, la chronoampérométrie ...
- Reproduction de diverses conditions expérimentales pour diverses molécules cibles (Isoproturon, Anthracène ...).
- Améliorer l'interprétation des réactions électrochimiques se produisant au sein du capteur.

Contact

aida.bensebaa@umontpellier.fr

This work was supported by a grant overseen by the French National Research Agency (ANR) as part of the "Investments d'Avenir" Programm.

[1] Pain Y. 2023. CARBON2023, Cancun.

[2] Pain Y. & Al. JE2024, St Malo

[3] Brown, J.H. 2015. Journal of Chemical Education.