

Transformation de la matière.
Étude d'une cinétique par conductimétrie.
 Compte rendu du TP.

* * *

Introduction

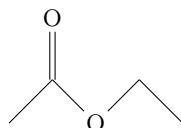


FIG. 1 – Représentation plane de l'éthanoate d'éthyle.

Au cours du précédent TP, nous avons pu effectuer un suivi cinétique de la réaction du cristal violet avec les ions hydroxydes afin de déterminer l'ordre global de la réaction. Cette fois-ci, nous mettrons en oeuvre une autre méthode expérimentale nous permettant de déterminer l'ordre globale d'une réaction. Il s'agit en l'occurrence de la réaction de saponification de l'acétate d'éthyle (ou éthanoate d'éthyle), et le suivi se fera par **conductimétrie**.



Éq. 1 – Équation de la réaction étudiée.

La réaction est totale. (1)

Nous réaliserons ainsi cette réaction et mesurerons la conductivité de la solution pour une durée de 20 minutes, le but étant, en connaissant la valeur de la conductivité initiale σ_0 et de la conductivité finale σ_∞ , de montrer que l'ordre global de cette réaction, puisqu'elle en admet un, est **2** et de déterminer, à la fin, l'énergie d'activation. Un travail préliminaire théorique nous permettra de mener à bien cette étude.

Préliminaires

Tout d'abord, la loi de vitesse de la réaction peut s'écrire :

$$v = k \times [\text{C}_4\text{H}_8\text{O}_2]^\alpha \times [\text{OH}^-]^\beta$$

où k est la constante de vitesse, α et β les ordres partiels. Or, les deux réactifs sont introduits en concentration égale, il s'ensuit que :

$$v = k \times [\text{OH}^-]^{\alpha+\beta} \quad (2)$$

On suppose à présente que $\alpha + \beta = 2$. La loi de vitesse peut aussi s'écrire :

$$v = -\frac{d[\text{OH}^-]}{dt}$$

Et on en déduit, d'après le relation (2), l'équation différentielle suivante :

$$\frac{d[\text{OH}^-]}{dt} = -k \times [\text{OH}^-]^2$$

En dissociant les variables et en intégrant entre t_0 et un temps quelconque t , on a :

$$\int_{[\text{OH}^-]_0}^{[\text{OH}^-]_t} \frac{-d[\text{OH}^-]}{[\text{OH}^-]^2} = k \times \int_0^t dt$$

Et il en résulte :

$$\boxed{\frac{1}{[\text{OH}^-]_t} = kt + \frac{1}{[\text{OH}^-]_0}} \quad (3)$$

Et on pose $C_0 = [\text{OH}^-] = [\text{C}_4\text{H}_8\text{O}_2]$.

Remarquons que $\frac{1}{[\text{OH}^-]_t} = f(t)$ est une fonction affine de t .

À présent, faisons une étude théorique de la réaction, notamment en nous intéressant à l'évolution de la conductivité en fonction du temps. D'après (1), la réaction est totale, ceci nous permet de dresser le tableau d'avancement suivant :

	$\text{C}_4\text{H}_8\text{O}_2$	OH^-	$\text{C}_2\text{H}_3\text{O}_2^-$	$\text{C}_2\text{H}_6\text{O}$
$t = 0$	C_0	C_0	0	0
t	$C_0 - x$	$C_0 - x$	x	x
$t = t_\infty$	0	0	C_0	C_0

Rappelons l'expression de la conductivité d'une solution, elle est égale à la somme des conductivités dues à chaque ion. Comme les espèces sont toutes monochargées, on a :

$$\boxed{\sigma = \sum_i \lambda_i \times [\text{X}_i]}$$

où λ_i est la conductivité molaire ionique de l'espèce X_i . On note dans ce cas σ_0 la conductivité à $t = 0$, σ la conductivité à un instant t quelconque, et enfin σ_∞ la conductivité de la solution à t_∞ . Remarquons que σ décroît au cours du temps : en effet, on assiste à une disparition d'ions hydroxydes et à une formation d'ions CH_3CO_2^- , or leur conductivité molaire ionique est inférieure à celle des ions OH^- ; ainsi, la somme des conductivités décroît. On donne les expressions des grandeurs que nous venons de définir :

$$\begin{cases} \sigma_0 = C_0 \times \lambda_{\text{OH}^-} + C_0 \times \lambda_{\text{Na}^+} \\ \sigma = (C_0 - x) \times \lambda_{\text{OH}^-} + x \times \lambda_{\text{C}_2\text{H}_3\text{O}_2^-} + C_0 \times \lambda_{\text{Na}^+} \\ \sigma_\infty = C_0 \times \lambda_{\text{C}_2\text{H}_3\text{O}_2^-} + C_0 \times \lambda_{\text{Na}^+} \end{cases} \quad (4)$$

Il en découle :

$$\frac{\sigma_0 - \sigma_\infty}{\sigma - \sigma_\infty} = \frac{C_0(\lambda_{\text{OH}^-} - \lambda_{\text{C}_2\text{H}_3\text{O}_2^-})}{(C_0 - x)(\lambda_{\text{OH}^-} - \lambda_{\text{C}_2\text{H}_3\text{O}_2^-})} = \frac{C_0}{C_0 - x}$$

Autrement dit :

$$\boxed{\frac{\sigma_0 - \sigma_\infty}{\sigma - \sigma_\infty} = \frac{C_0}{[\text{OH}^-]_t}} \quad (5)$$

Ainsi, mesurer la conductivité de la solution au cours du temps nous permettra de tracer la droite $\frac{1}{[\text{OH}^-]_t} = f(t)$ à une constante près. En effet, d'après (3) :

$$\frac{C_0}{[\text{OH}^-]_t} = 1 + C_0 kt \quad (6)$$

Expérienc et résultats

On travaille à une température de $T_1 = 294,15 \text{ K}$. Comme il est difficile de mesurer la conductivité initiale après avoir lancé la réaction, on dilue 50mL de soude à $0,05 \text{ mol.L}^{-1}$ dans 50mL d'eau distillée, et on mesure σ_0 après agitation. De la même manière, pour ne pas attendre trop longtemps, on dilue 50mL d'éthanoate de sodium à $0,05 \text{ mol.L}^{-1}$ dans 50mL d'eau distillée, on mesure σ_∞ après agitation.

σ_0	4,94mS.cm ⁻¹
σ_∞	1,0225mS.cm ⁻¹

On verse ensuite 50mL de soude à $0,05 \text{ mol.L}^{-1}$ dans un bécher de 150mL, on y ajoute 50mL d'éthanoate d'éthyle à $0,05 \text{ mol.L}^{-1}$; on déclenche le chronomètre dès que la totalité du volume a été versée. On note la conductivité de la solution chaque 30s pendant 20 minutes. Le tableau de valeurs est disponible en annexe.

Calculs numériques

Dans cette sous-partie, nous utiliserons un programme Python pour effectuer les calculs numériques. Nous reprendrons les fonctions du fichier `parser.py` qu'on a utilisées dans le précédent compte-rendu, que nous importerons dans un fichier que nous nommerons cette fois-ci `main.py`, le programme est toujours disponible en annexe. On trace alors $\frac{C_0}{[\text{OH}^-]_t}$ en fonction de t , et on regarde les résultats qu'on obtient en console.

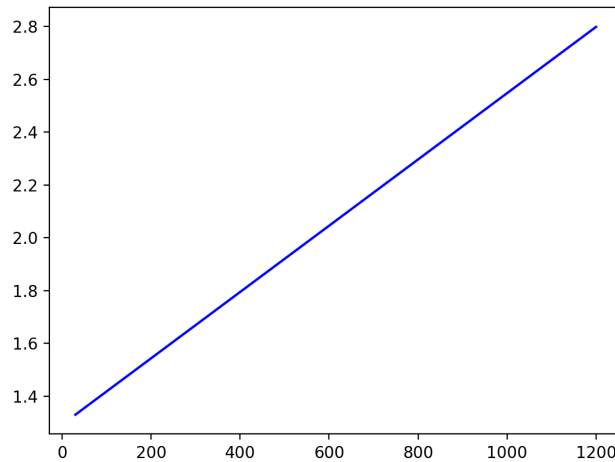


FIG 2. $-\frac{C_0}{[\text{OH}^-]_t} = f(t)$

Les résultats en console donnent :

```
tiplouf:cmeter mac$ p3 main.py
r**2 = 0.9907695417208003
a = C0*k = [0.00125436] b = 1.293032827451996
```

L'ordre 2 est donc bien validé. De plus, le coefficient directeur étant $C_0 \times k$, on peut en déduire la valeur de k , on a

$$k(T_1) = \frac{a}{C_0} = 5,01744 \times 10^{-2} \text{ L.mol}^{-1}\text{s}^{-1} \quad (7)$$

En sachant qu'à $T_2 = 277,15\text{K}$, $k(T_2) = 1,06 \times 10^{-2} \text{ L.mol}^{-1}\text{s}^{-1}$, on peut en déduire l'énergie d'activation E_a . En effet, la loi d'Arrhenius permet d'établir :

$$\begin{aligned} \begin{cases} k(T_1) = A \times \exp\left(\frac{-E_a}{RT_1}\right) \\ k(T_2) = A \times \exp\left(\frac{-E_a}{RT_2}\right) \end{cases} &\Leftrightarrow \frac{k(T_1)}{k(T_2)} = \exp\left(\frac{E_a}{RT_2} - \frac{E_a}{RT_1}\right) \\ &\Leftrightarrow \ln\left(\frac{k(T_1)}{k(T_2)}\right) = E_a \times \left(\frac{1}{RT_2} - \frac{1}{RT_1}\right) \\ &\Leftrightarrow E_a = \frac{RT_1 T_2 \ln\left(\frac{k(T_1)}{k(T_2)}\right)}{T_1 - T_2} \end{aligned}$$

Application numérique : $E_a = 61980.23 \text{ J.mol}^{-1}$

Annexe

Données :

Ci-dessous, le tableau contenant les valeurs de la conductivité mesurées à intervalles réguliers pendant 20 minutes.

i	t	σ	i	t	σ
	s	S/cm		s	S/cm
0	30,00	4,110	20	630,0	2,890
1	60,00	4,010	21	660,0	2,860
2	90,00	3,900	22	690,0	2,830
3	120,0	3,800	23	720,0	2,810
4	150,0	3,700	24	750,0	2,780
5	180,0	3,630	25	780,0	2,760
6	210,0	3,540	26	800,0	2,680
7	240,0	3,480	27	810,0	2,740
8	270,0	3,410	28	830,0	2,660
9	300,0	3,350	29	840,0	2,710
10	330,0	3,290	30	860,0	2,640
11	360,0	3,240	31	870,0	2,700
12	390,0	3,190	32	890,0	2,620
13	420,0	3,140	33	920,0	2,610
14	450,0	3,100	34	950,0	2,590
15	480,0	3,060	35	980,0	2,580
16	510,0	3,020	36	1010	2,560
17	540,0	2,980	37	1040	2,550
18	570,0	2,950	38	1070	2,512
19	600,0	2,920	39	1200	2,502

Programme informatique

Voici le fichier main.py

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3 from sklearn.linear_model import LinearRegression
4 from parser import *
5
6 f = open("data.txt", 'r')
7 model = LinearRegression()
8
9 ~ if __name__ == "__main__":
10     clean_lines = remove_junk(f)
11     final = final_data(clean_lines)
12     n = len(final)
13     t = np.array([final[i][1] for i in range(n)])
14     t1 = t.reshape((-1, 1))
15     s = np.array([final[i][2] for i in range(n)])
16     c = 3.9175/(s-1.0225)
17
18     new_model = model.fit(t1, c)
19     print(f'r^2 = {model.score(t1, c)}')
20     a, b = new_model.coef_, new_model.intercept_
21     print(f'a = -k = {a}', f'b = {b}')
22     y = a*t + b
23     plt.plot(t, y, 'b-')
24     plt.show()

```