Numerical Analysis For Materials

Homework #9

Gilwoon Lee

ID: 20120083
Dept: Material Science and Engineering

POSTECH

Homework #9

Partial 2nd ODE: Diffusion

1. Theory

3. Finite Difference Methods for Parabolic Problems

- Heat or diffusion equation

$$\frac{\partial u}{\partial t}(x,t)=\alpha\,\frac{\partial^2 u}{\partial x^2}(x,t) \qquad \text{for} \quad 0<x<l \quad \text{and} \quad t>0$$

subject to the condition

$$u(0,t) = u(l,t) = 0 \qquad \text{for} \quad t>0$$
$$u(x,0) = f(x) \qquad \text{for} \quad 0 \le x \le l$$

set $x_i = i\Delta x$ for i = 0, 1, 2, ⋯, m

$t_j = j\Delta t$ for j = 0, 1, 2, ⋯

$$\frac{\partial u}{\partial t}(x_i,t_j)=\frac{u(x_i,t_j+\Delta t)-u(x_i,t_j)}{\Delta t}-\frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_j)$$

$$\frac{\partial^2 u}{\partial x^2}(x_i,t_j)=\frac{u(x_i+\Delta x,t_j)-2u(x_i,t_j)+u(x_i-\Delta x,t_j)}{\Delta x^2}-\frac{\Delta x^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i,t_j)$$

$$\frac{w_{i,j+1}-w_{i,j}}{\Delta t}=\alpha\,\frac{w_{i+1,j}-2w_{i,j}+w_{i-1,j}}{\Delta x^2}$$

$$w_{i,j+1}=w_{i,j}+\frac{\alpha\Delta t}{\Delta x^2}(w_{i+1,j}-2w_{i,j}+w_{i-1,j})=\left(1-2\frac{\alpha\,\Delta t}{\Delta x^2}\right)w_{i,j}+\frac{\alpha\,\Delta t}{\Delta x^2}(w_{i+1,j}+w_{i-1,j})$$

$$A=\begin{bmatrix}(1-2\lambda) & \lambda & 0 & \cdots & 0\\ \lambda & (1-2\lambda) & \lambda & \ddots & \vdots\\ 0 & \ddots & \ddots & \ddots & 0\\ \vdots & & \ddots & \ddots & \lambda\\ 0 & \cdots & 0 & \lambda & (1-2\lambda)\end{bmatrix} \qquad \lambda=\frac{\alpha\Delta t}{\Delta x^2}$$

$$\mathbf{w}^{(0)}=(f(x_1), f(x_2), \ldots, f(x_{m-1}))^t$$

$$\mathbf{w}^{(j)}=(w_{1j}, w_{2j}, \ldots, w_{m-1,j})^t, \quad \text{for each } j=1,2,\ldots,$$

$$\mathbf{w}^{(j)}=A\mathbf{w}^{(j-1)}, \quad \text{for each } j=1,2,\ldots.$$

⇒ Forward Difference Method (Explicit Method)

∴ Stability condition of explicit method: $\dfrac{\alpha\Delta t}{\Delta x^2}\le\dfrac{1}{2}$

ex) for a diffusion controlled reaction, $t_f \propto 1\,/\,D$

$$D = D_o \cdot exp\ (-Q^{diff}/RT)$$

$$Q^{reac} = R\cdot\frac{\partial(\ln 1/D)}{\partial(1/T)} = -R\frac{\partial(-Q^{diff}/RT)}{\partial(1/T)} = Q^{diff}$$

1. Theory

$$\frac{\partial u}{\partial t}(x,t) = D\,\frac{\partial^2 u(x,t)}{\partial x^2}$$

(where $D = 4.529 \times 10^{-7}\exp(-147723(J)/RT)$)

$$\therefore \lambda = \frac{D\Delta t}{(\Delta t)^2}$$ (where $\Delta x = 1\ \mu m$, $\Delta t = 1\ sec$)

If $\lambda > \frac{1}{2}$, add $0.5\ \mu m$ to Δx until $\lambda \le \frac{1}{2}$ can be satisfied

$$A = \begin{bmatrix} (1-2\lambda) & \lambda & 0 & \cdots & 0 \\ \lambda & (1-2\lambda) & \lambda & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \lambda \\ 0 & \cdots & 0 & \lambda & (1-2\lambda) \end{bmatrix}$$

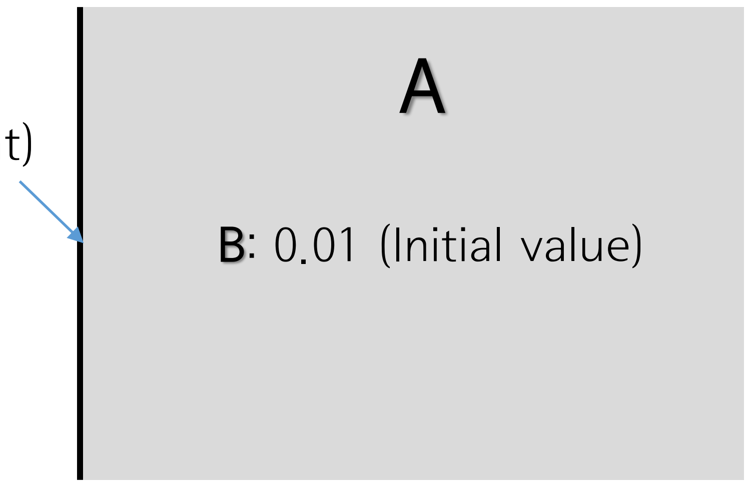
$$\mathrm{w}^{(0)} = (0.05, 0.01, 0.01, \ldots, 0.01)^{\mathrm{T}}$$
$$\mathrm{w}^{(t)} = \mathrm{A}\mathrm{w}^{(t-1)}$$

Injection distance: where concentration becomes 0.03

Concentration and time difference can not be less than given increment, therefore **Interpolation & Regression (Homework #5)** is used to estimate values.

Problem situation

B: 0.05 (constant)



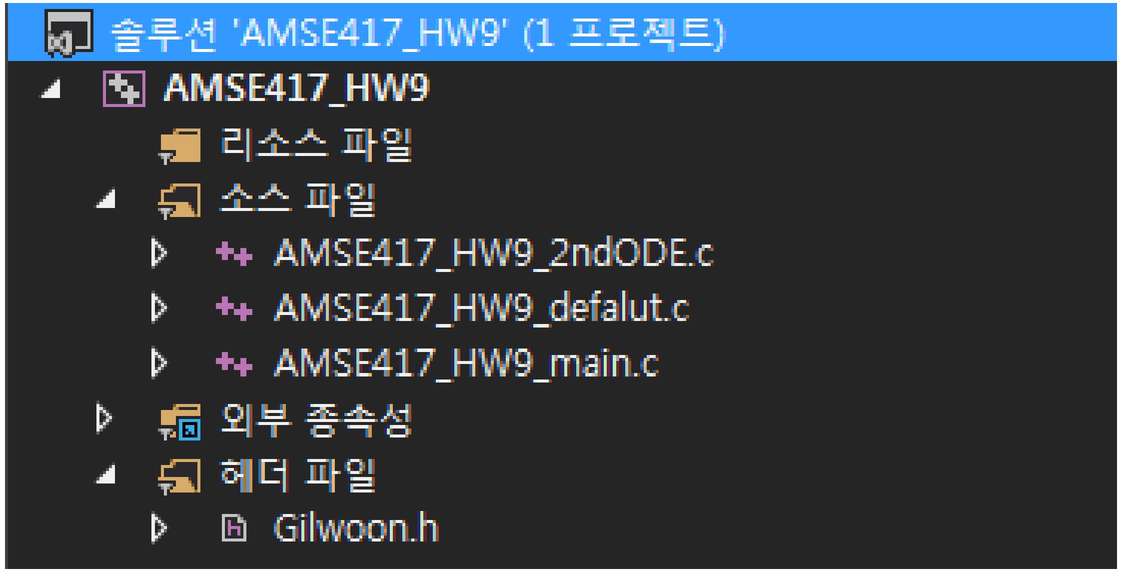
$T = 1173\ K \sim 1473\ K$ **Target value of B:** 0.03

$D = 4.529 \times 10^{-7}\exp(-147723(J)/RT)$

2. Programmed code

* Description of Program

1. Info. of Program



2. Characteristics of Program

- 헤더 및 각 기능 함수화
- Increment 수정 용이
- 각 온도를 txt 파일명으로 하여 data 출력
- 출력 파일은 50 K 간격으로 생성

2. Programmed code

* Description of Program

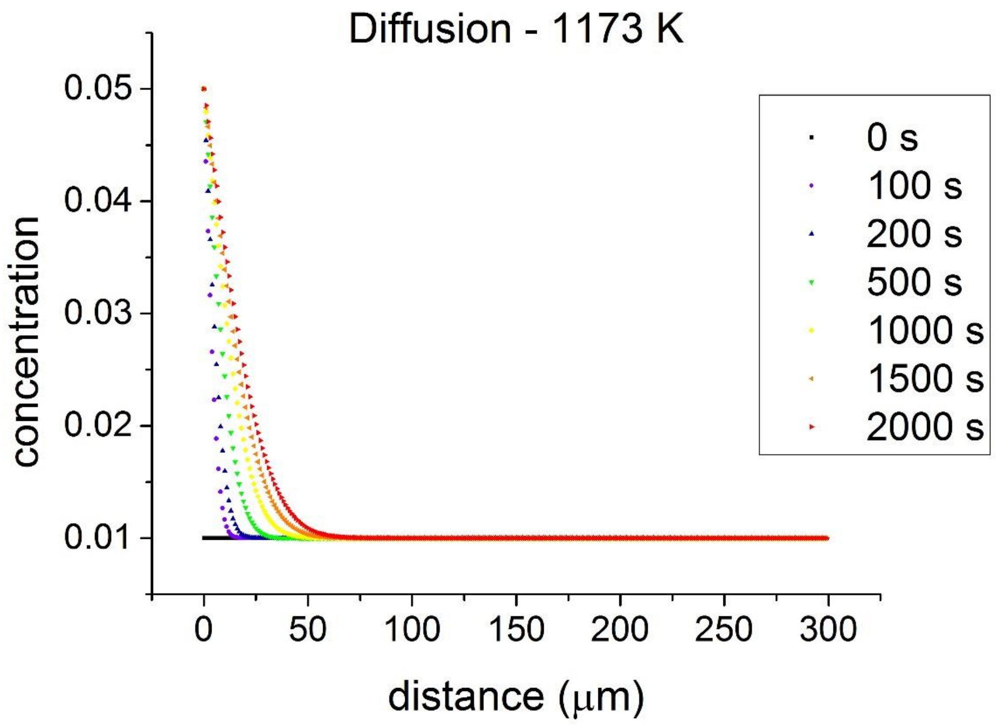
3. void _2ndODE()

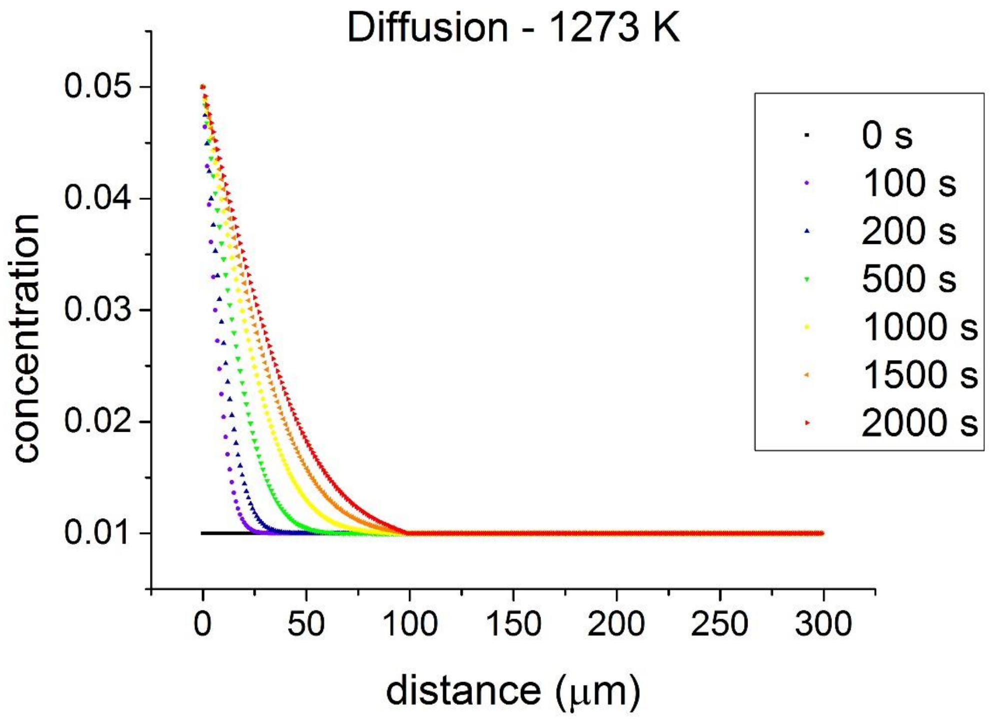
```
void _2ndODE()
{
FILE *file;
char filename[100];
double W[100];
double W_temp[100];
double A[100][100] = { 0, };
double dx = 2.5*pow(10,-6);
double dt = 1;
int t = 0;
int i, j, k, l;
double T, D, lambda;

for (k = 0; k < 100; k++) { W[k] = 0.01; W_temp[k] = 0; }

//T, D, lambda 값 구하기
T = 1473;
D = 4.529 * pow(10, -7) * exp(-147723 / (R*T));
lambda = D*dt / pow(dx, 2);

//A 값 할당, 나머지 숫자는 0으로 초기화 되어있음
for (k = 0; k < 100; k++) { A[k][k] = 1 - 2 * lambda; }
for (k = 1; k < 100; k++) { A[k][k - 1] = lambda; A[k - 1][k] =
lambda; }
```

```
//W 값 할당
W[0] = 0.05;
W_temp[0] = 0.05;
j = 0;
_itoa(t, filename, 10);
strcat(filename, ".txt");
file = fopen(filename, "w");
fprintf(file, "%d\n", t);
for (k = 0; k < 100; k++) { fprintf(file, "%f\n", W[k]); }
fclose(file);
t += dt;
while (t <= 2500)
{
for (k = 1; k < 99; k++){ for (i = 0; i < 100; i++) { W_temp[k]
+= A[k][i] * W[i]; } }
for (k = 1; k < 99; k++) { W[k] = W_temp[k]; }
if (t %50==0)
{
_itoa(t, filename, 10);
strcat(filename, ".txt");
file = fopen(filename, "w");
fprintf(file, "%d\n", t);
for (k = 0; k < 100; k++) { fprintf(file, "%f\n", W[k]); }
fclose(file);
}
for (k = 1; k < 99; k++) { W_temp[k] =0; }
t += dt;
}
}
```

3. Result & Conclusion

1. Diffusion simulation result



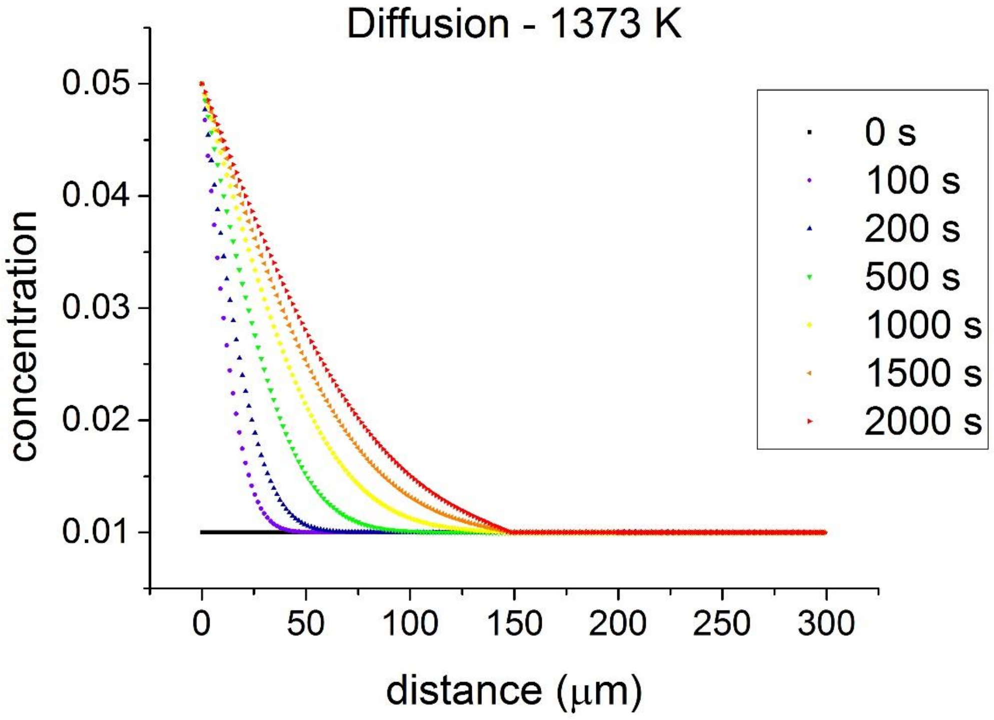


$\Delta x = 1\ \mu m,\ \Delta t = 1\ sec$
$\lambda = 0.11964802$

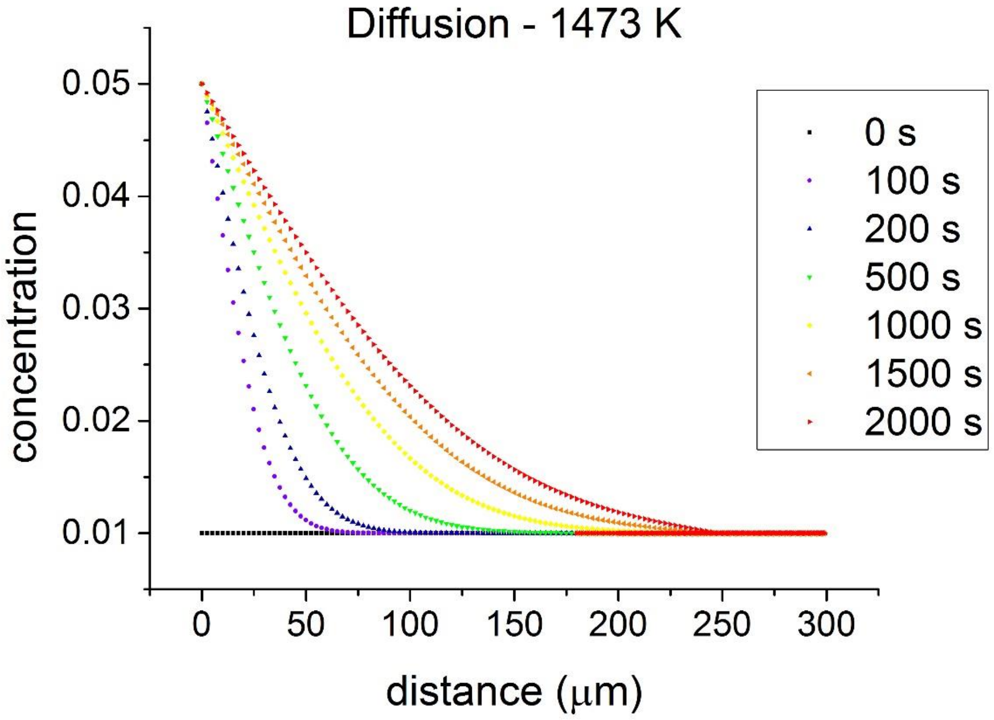
$\Delta x = 1\ \mu m,\ \Delta t = 1\ sec$
$\lambda = 0.39322874$

3. Result & Conclusion

1. Diffusion simulation result



$$\Delta x = 1.5\ \mu m,\ \Delta t = 1\ sec$$
$$\lambda = 0.48298188$$



$$\Delta x = 2.5\ \mu m,\ \Delta t = 1\ sec$$
$$\lambda = 0.41856228$$

3. Result & Conclusion

2. Interpolation method (HW#5): Lagrange 방법 이용

주어진 data 중 0.03 주변에 있는 5~6개의 data를 뽑아서 0.03에 해당되는 Injection distance를 구함.

```
Info - 메모장
파일(F)  편집(E)
6
0.022318 5
0.026585 4
0.031623 3
0.037331 2
0.043534 1
0.05 0
```

```
C:\Windows\system32\cmd.exe
==========================================================================
Program: Interpolation of given data - Lagrange
Date: 2015.04.13
Made by Gilwoon Lee
POSTECH, project 4 of [AMSE417] Numerical analysis for materials
Development environment: Visual Studio 2013
Code language: C
==========================================================================

This programm will do interpolation of given data.
============================
To get f(x), enter x: 0.03
Range is 0.026585 < x < 0.031623
The answer is: 3.306596
```

	100 s	200 s	500 s	1000 s	1500 s	2000 s
0	0.05	0.05	0.05	0.05	0.05	0.05
0	0.05	0.05	0.05	0.05	0.05	0.05
1	0.043534	0.045407	0.047087	0.047939	0.048316	0.048542
2	0.037331	0.040909	0.044199	0.045886	0.046638	0.047087
3	0.031623	0.036595	0.041359	0.04385	0.044968	0.045637
4	0.026585	0.032542	0.038589	0.04184	0.043312	0.044197
5	0.022318	0.028813	0.035909	0.039863	0.041675	0.042769
6	0.018851	0.025454	0.03334	0.037927	0.04006	0.041356
7	0.016147	0.022489	0.030896	0.036039	0.038472	0.039961
8	0.014124	0.019926	0.02859	0.034205	0.036915	0.038585
9	0.012672	0.017756	0.026434	0.032431	0.035392	0.037233
10	0.01167	0.015957	0.024433	0.030723	0.033907	0.035905
11	0.011007	0.014496	0.022593	0.029085	0.032463	0.034605
12	0.010586	0.013334	0.020915	0.02752	0.031062	0.033335
13	0.010328	0.012428	0.019396	0.026032	0.029707	0.032096
14	0.010178	0.011737	0.018033	0.024623	0.0284	0.03089
15	0.010093	0.01122	0.016821	0.023293	0.027144	0.029719
16	0.010046	0.010841	0.015751	0.022044	0.025938	0.028584
17	0.010023	0.010569	0.014815	0.020876	0.024786	0.027487
18	0.010011	0.010378	0.014003	0.019788	0.023686	0.026427
19	0.010005	0.010247	0.013304	0.018779	0.02264	0.025407
20	0.010002	0.010158	0.012707	0.017847	0.021649	0.024427
21	0.010001	0.010099	0.012203	0.016989	0.02071	0.023486
22	0.01	0.010061	0.011779	0.016204	0.019826	0.022586
23	0.01	0.010037	0.011426	0.015487	0.018993	0.021726
24	0.01	0.010022	0.011135	0.014836	0.018213	0.020907

0.028500~0.031500에 해당하는 data를 분홍색으로 표시하여 0.03 근처를 알아보기 쉽게 만듦

3. Result & Conclusion

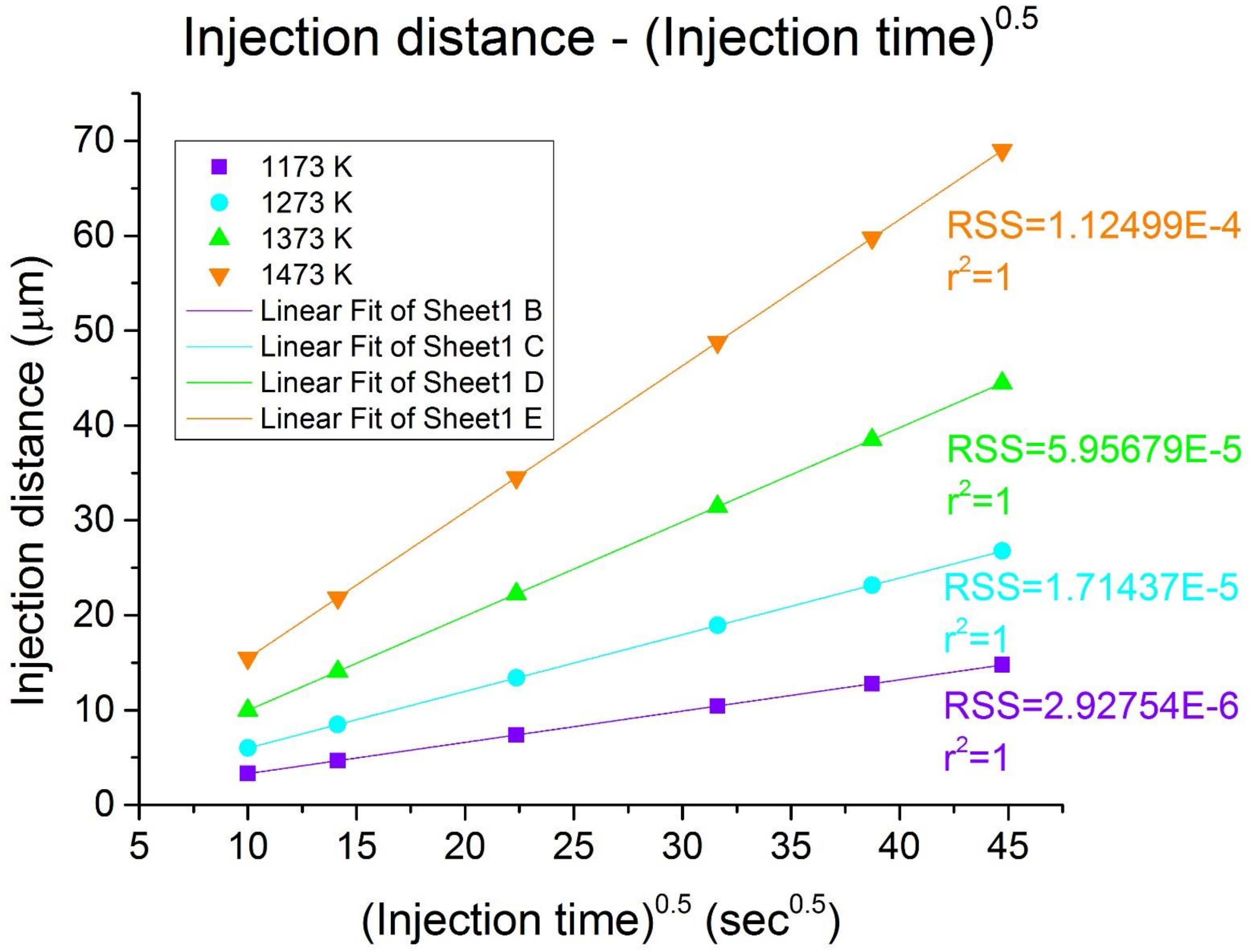
3. Injection distance data

	1173 K	1273 K	1373 K	1473 K
100 sec	3.306596	5.998951	9.97284	15.472735
200 sec	4.671367	8.471233	14.083246	21.849882
500 sec	7.381163	13.383106	22.247654	34.517476
1000 sec	10.436037	18.92049	31.45412	48.801013
1500 sec	12.780801	23.171263	38.517948	59.762816
2000 sec	14.75726	26.753151	44.461978	69.002887

(단위: μm)

3. Result & Conclusion

Problem (a) Injection distance - Injection time



$$u(x,t) = u_0 \mathrm{erfc}\left(\frac{x}{2\sqrt{Dt}}\right)$$

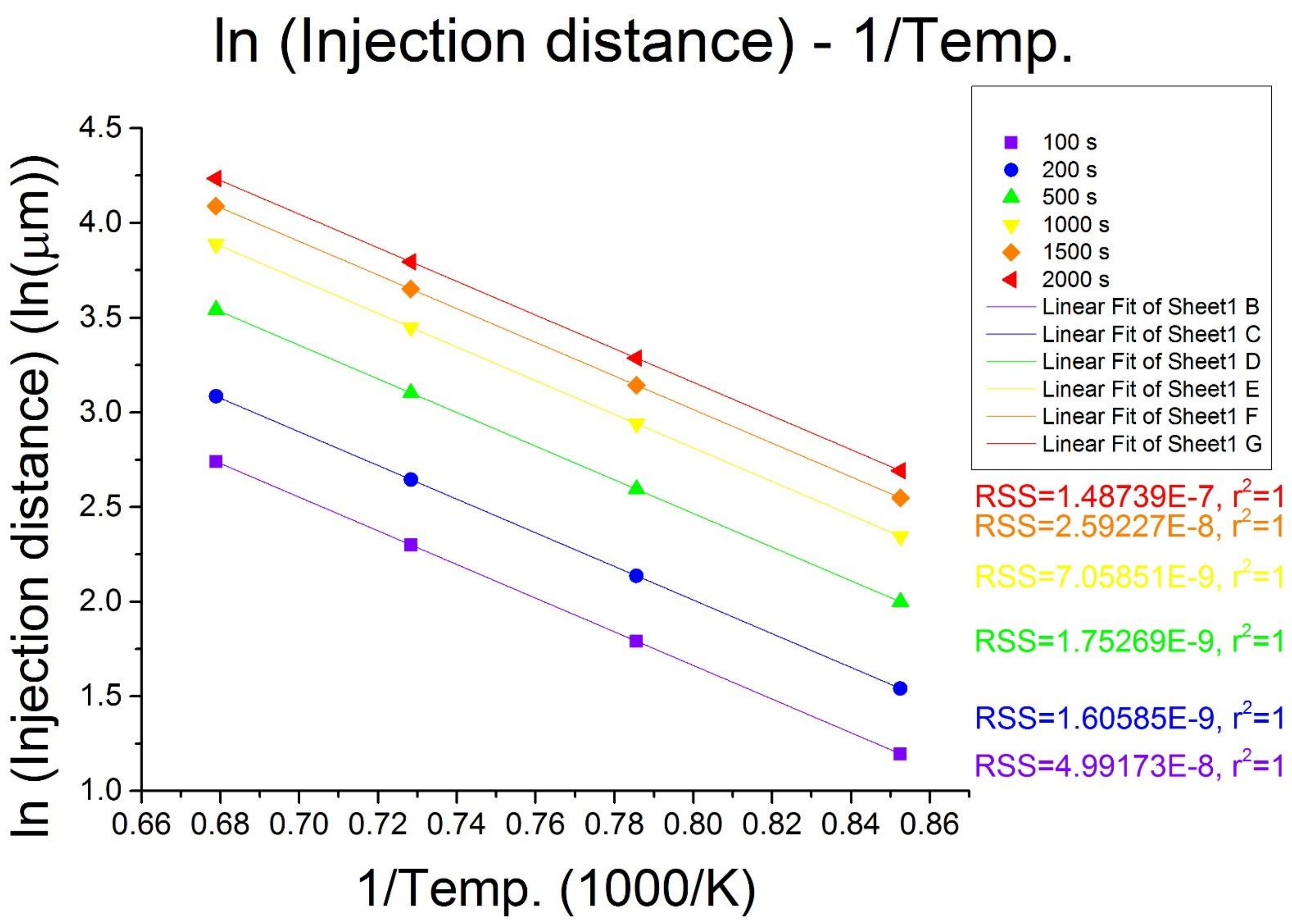
For fixed $u = 0.03$, $u_0 = 0.05$, and D for each T,

$$x \propto \sqrt{t}$$

For each temperature, <u>injection distance</u> shows linear proportional to <u>square root of injection time</u>.

3. Result & Conclusion

Problem (b) Injection distance - Temperature



$$u(x,t) = u_0 \mathrm{erfc}(\frac{x}{2\sqrt{Dt}})$$

$$D = 4.529 \times 10^{-7} \exp(-147723(J)/RT)$$

For fixed $u = 0.03$, $u_0 = 0.05$, each t,

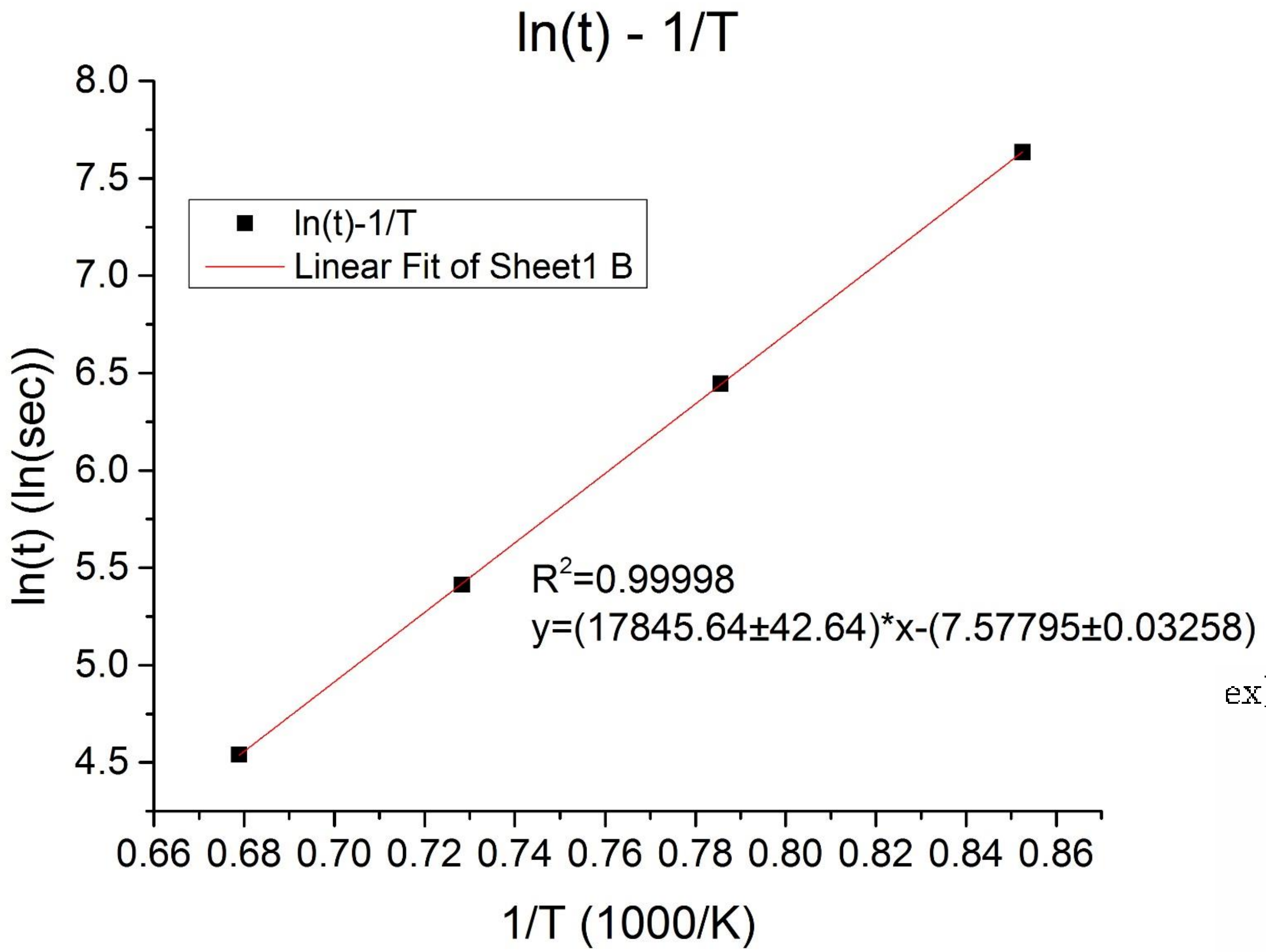
$$\ln x \propto \frac{1}{T}$$

For each time, <u>ln(injection distance)</u> shows linear proportional to <u>inverse of temperature</u>.

3. Result & Conclusion

Problem (c) activation energy: Injection time - Temp.

$$Q^{reac} = R \times \frac{\partial(\ln t_f)}{\partial(1/T)}$$



For fixed $\mathrm{d} = 15\ \mu m,\ u_0 = 0.03$

$$\partial(\ln t_f) = \frac{Q^{reac}}{R} \times \partial(1/T)$$

(Cubic spline used for estimation of each data)

$$Q^{reac} = 17845.64 \times 8.3144621\ (J) = 148376.90\ (J)$$

$$\frac{148376.90 - 147723}{147723} \times 100(\%) = 0.44\ \%$$

ex) for a diffusion controlled reaction, $t_f \propto 1 / D$

$$D = D_o \cdot exp\ (-Q^{diff}/RT)$$

Diffusion process!!

$$Q^{reac} = R \cdot \frac{\partial(\ln 1/D)}{\partial(1/T)} = -R\frac{\partial(-Q^{diff}/RT)}{\partial(1/T)} = \boxed{Q^{diff}}$$

3. Result & Conclusion

Problem (c) activation energy: Injection time - Temp. (Supplementary data)

각 온도의 좌측 data: injection distance, 우측 data: injection time

1173 K		1273 K		1373 K		1473 K	
0.01	0	0.01	0	0.01	0	0.01	0
0.010093	100	0.013645	100	0.022405	100	0.024249	50
0.01122	200	0.019288	200	0.0289	200	0.030528	100
0.016821	500	0.027986	500	0.035972	500	0.035735	200
0.023293	1000	0.033714	1000	0.039909	1000	0.040778	500
0.027144	1500	0.036495	1500	0.041713	1500	0.043431	1000
0.029719	2000	0.038212	2000	0.042802	2000	0.044623	1500
0.031589	2500					0.045337	2000
0.03	2068.810637	0.03	629.542433	0.03	224.466749	0.03	93.667424

4. Conclusion

1) Diffusion process의 simulation을 성공적으로 수행하였고, (a), (b)에서 각각 이론적으로 $x \propto \sqrt{t},\ \ln x \propto \frac{1}{T}$의 비례 결과가 나옴을 simulation을 통해서 다시 한 번 확인하였다.
2) 주어진 reaction의 reaction activation energy(Q^{reac})를 구하여 값을 비교한 결과, diffusion mechanism에 해당함을 알 수 있었다.
3) Origin을 사용하여 깔끔한 그래프를 그리는 연습을 할 수 있었다.