Matlab Project (PHE1024/FTE1023)

By Martin Wijaya Hermanto

Self Introduction

Education

- **2009 – PhD in Chemical and Biomolecular Engineering (NUS)**

  Thesis: Modelling, Simulation and Control of Polymorphic Crystallization

- **2004 – B. Eng in Chemical and Biomolecular Engineering (NUS)**

  FYP: Quantitative measures and guidelines for upgrading simple feedback to cascade control

Self Introduction

Working Experience

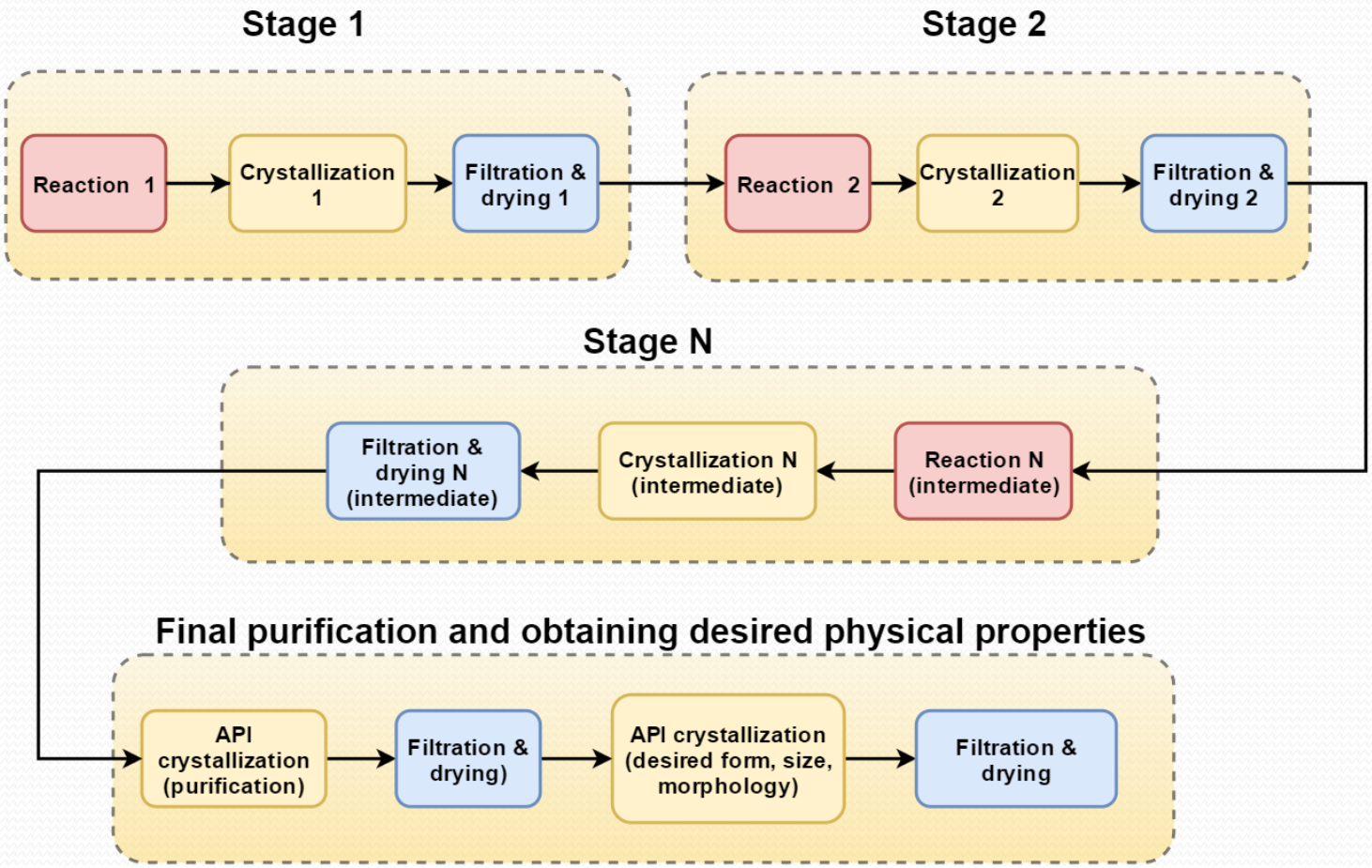
- **GSK**, Senior Chemometrician I (Sep 2015 – present)
- **Institute of Chemical & Engineering Sciences**, Scientist II (Aug 2008 – Sep 2015)

PHE1024/FTE1023 Matlab Project

- Basic understanding and importance of crystallization process in pharmaceutical industries

- Write Matlab program to:
  1. Simulate crystallization process
  2. Estimate kinetic parameter
  3. Design cooling rate to satisfy desired conditions

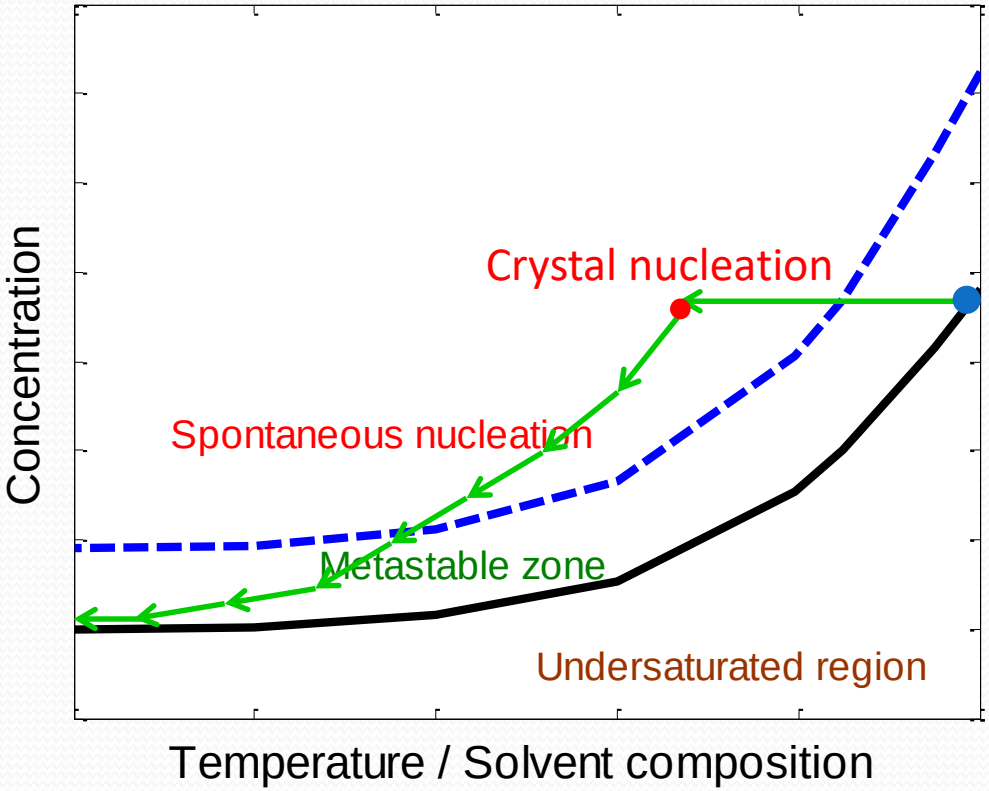
Example of pharmaceutical primary manufacturing process (API formation stage)



Importance of crystallization process

- Majority of pharmaceutical manufacturing process involve a series of crystallization processes
- Purpose:
  - Isolation of intermediates
  - Purification and separation
  - Obtaining the desired crystal form (polymorph)
  - Obtaining desired morphology (shape) and size
- Common types of crystallization in pharmaceutical industries:
  - Cooling crystallization (driven by solubility change due to temperature)
  - Antisolvent crystallization (driven by solubility change due to addition of solvent with less solubility)
- Driving Force:
  - Supersaturation → the difference between solute concentration and its solubility in the current solvent

Solubility curve and crystallization progression



Phase transitions:

- Undersaturated region: No nucleation or growth, only dissolution
- Metastable zone: No nucleation, only growth
- Spontaneous nucleation: Nucleation and growth

How to simulate a batch cooling crystallization

- The c… Populati… a parti…

- Coupled with … integro-differ… equa…

$$\frac{dC'}{dt} = -\frac{3\rho_c k_v}{\alpha V_{slurry}} \int_0^{\infty} L^2 G\, f\, d$$

$$C'(0) = C_0'$$

Don't Worry!!!!
This is a subject for Master/PhD Degree

Simplifying the PBE

- For certain conditions, the population balance can be simplified into a set of ordinary differential equations (ODEs):

$$\frac{d\mu_0}{dt} = B \quad \ldots(1)$$

$$\frac{d\mu_1}{dt} = G\mu_0 \quad \ldots(2)$$

$$\frac{d\mu_2}{dt} = 2G\mu_1 \quad \ldots(3)$$

$$\frac{d\mu_3}{dt} = 3G\mu_2 \quad \ldots(4)$$

$$\frac{d\mu_4}{dt} = 4G\mu_3 \quad \ldots(5)$$

where μ are the moments, B is the nucleation kinetic equation, G is the growth kinetic equation. μ_o to μ_3 are proportional to number, size, surface area, and volume of crystals, respectively

Simplifying the solute mass balance

- The solute mass balance becomes another ODE:

$$\frac{dC}{dt} = -\frac{3\rho_{cryst} k_v G \mu_2}{\rho_{solv}} \qquad \ldots (6)$$

where C is the solute concentration, ρ_{cryst} is the density of crystal, ρ_{solv} is the density of solvent and k_v is the volume shape factor of the crystal (assume spherical crystal in this problem)

Kinetics and solubility expressions

- The nucleation and growth kinetics equations are given by:

$$B = \begin{cases} k_b \left(C - C_{sat}\right)^b & if\ C > C_{sat} \\ 0 & if\ C \leq C_{sat} \end{cases}$$

$$G = \begin{cases} k_g \left(C - C_{sat}\right)^g & if\ C > C_{sat} \\ 0 & if\ C \leq C_{sat} \end{cases}$$

where kb and b are the nucleation kinetic parameters, while kg and g are the growth kinetic parameters. C_{sat} is the saturation concentration (solubility) equation as a function of temperature

- The solubility equation C_{sat} is given by:

$$C_{sat} = 6.59033 \times 10^{-3} \exp\left(3.27088 \times 10^{-2} T\right)$$

where T is the temperature

Temperature profile

- The temperature profile follows a linear expression:

$$T(t) = T_0 - \beta t$$

where T_o is the initial temperature and β is the cooling rate.

- The temperature cannot go below a minimum temperature, that is:

$$T(t) \geq 5$$

Parameter values

Parameters	Values
k_g	exp(-17) (for task 1 only, otherwise use the estimated k_g obtained in task 2, which is different for each group)
g	1.5
k_b	Custom for each group
b	6.0
ρ_{cryst}	1.263×10^3
ρ_{solv}	1.0×10^3
k_v	$4/3 \times \pi$
β	15.0 (for tasks 1 and 2)

Initial values for the 6 ODEs

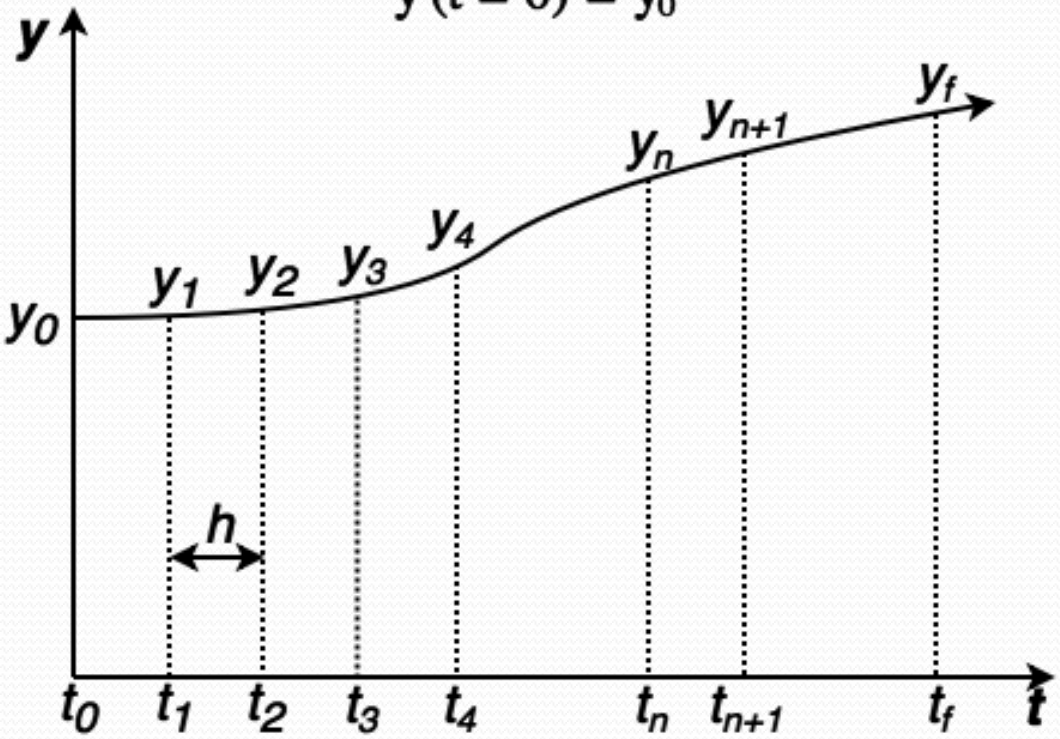
Variables	Values at time = 0
$\mu_0(t=0)$ to $\mu_4(t=0)$	0
$C(t=0)$	Saturated with respect to initial temperature (the initial temperature is custom for each group)
$T(t=0)$ or T_0	Custom for each group

Ordinary differential equation with initial value problem

Ordinary Differential Equation (ODE) with initial value

$$\frac{dy}{dt} = f(t, y)$$

$$y(t = 0) = y_0$$



Solving the ODE using 4th order Runge-Kutta Method

- Classical Runge-Kutta 4th order method to solve ODEs with initial values problem:

$$
\begin{aligned}
k_1 &= h\, f\left(t_n, y_n\right) \\
k_2 &= h\, f\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \\
k_3 &= h\, f\left(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right) \\
k_4 &= h\, f\left(t_n + h, y_n + k_3\right) \\
t_{n+1} &= t_n + h \\
y_{n+1} &= y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}
\end{aligned}
$$

(a) Main or "Driver" Program

```
Assign values for
y = initial value dependent variable
xi = initial value independent variable
xf = final value independent variable
dx = calculation step size
xout = output interval

x = xi
m = 0
xp_m = x
yp_m = y
DO
   xend = x + xout
   IF (xend > xf) THEN xend = xf
   h = dx
   CALL Integrator (x, y, h, xend)
   m = m + 1
   xp_m = x
   yp_m = y
   IF (x ≥ xf) EXIT
END DO
DISPLAY RESULTS
END
```

(b) Routine to Take One Output Step

```
SUB Integrator (x, y, h, xend)
  DO
    IF (xend − x < h) THEN h = xend − x
    CALL Euler (x, y, h, ynew)
    y = ynew
    IF (x ≥ xend) EXIT
  END DO
END SUB
```

***(c)* Routine of 4th order RK Method**

```
SUB RK4 (x, y, h, ynew)
  CALL Derivs(x, y, k1)
  ym = y + k1 · h/2
  CALL Derivs(x + h/2, ym, k2)
  ym = y + k2 · h/2
  CALL Derivs(x + h/2, ym, k3)
  ye = y + k3 · h
  CALL Derivs(x + h, ye, k4)
  slope = (k1 + 2(k2 + k3) + k4)/6
  ynew = y + slope · h
  x = x + h
END SUB
```

(d) Routine to Determine Derivative

```
SUB Derivs (x, y, dydx)
  dydx = . . .
END SUB
```

Solving the ODE using 4th order Runge-Kutta Method

- For our case, the *y* and *f(t,y)* are vectors given by:

$$y = \begin{bmatrix} \mu_0 \\ \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \\ C \end{bmatrix}$$

$$f(t,y) = \begin{bmatrix} B \\ G\mu_0 \\ 2G\mu_1 \\ 3G\mu_2 \\ 4G\mu_3 \\ -\frac{3\rho_{cryst}k_v G\mu_2}{\rho_{solv}} \end{bmatrix}$$

Solving the ODE using 4th order Runge-Kutta Method

- The initial conditions y_0 is given by:

$$y_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ C_0 \end{bmatrix}$$

where C_0 is the initial concentration, which is saturated with respect to initial temperature T_o . This value is obtained from

$$C_0 = 6.59033 \times 10^{-3} \exp\left(3.27088 \times 10^{-2} T_0\right)$$

- Note that the initial temperature T_0 is different for each group

ODE Parameter values

Parameters	Values
h	60 seconds
t_f	10 hours

Task 1 – Solve the ODEs

All required parameters are listed in previous tables, except for the custom parameters/initial temperature for each group are given below:

Group	k_b	T_o
1	45.0	55.8
2	42.4	59.4
3	42.6	59.7
4	43.6	58.9
5	42.8	57.2
6	42.4	59.0
7	44.7	57.4
8	43.9	57.4
9	42.4	56.3
10	42.7	57.5
11	44.4	58.0

Group	k_b	T_o
12	43.5	55.5
13	44.4	56.7
14	44.5	56.2
15	43.6	58.6
16	43.4	56.7
17	43.7	55.9
18	43.3	57.4
19	42.6	57.4
20	42.8	55.9
21	44.3	57.2
22	44.8	56.5

Task 1 – Solve the ODEs

- Using cooling rate of 15 ^{o}C/hr, solve the ODEs using the given Runge-Kutta method for time t_0 to t_f with time step h. *Note that use seconds for time unit when solving the ODEs, as the kinetics parameters are in seconds. You can convert to hours after solving the ODE.*

- After solving the ODE, plot the following profiles (versus *time in hrs*) and give appropriate legends:
  - Figure 1 – Temperature profile
  - Figure 2 – Concentration (C) and saturation concentration (C_{sat}) profile
  - Figure 3 – Subplot μ_0 to μ_2 profile
  - Figure 4 – Subplot μ_3 to μ_4 profile

Hints for Task 1

- Write a function to calculate the saturation concentration (C_{sat}) as a function of temperature (T)
- Write functions for both growth (G) and nucleation (B) kinetics as functions of temperature (T) , concentration (C) , and the kinetic parameters (k_g, g, k_b, b)
- Write a function to calculate the outputs of function f.

  Note:
  - The outputs will be a vector
  - This function will need to call the kinetic functions above
- Write a Runge-Kutta function to calculate y_{n+1}:

  Note:
  - You need to give current t_n, $\boldsymbol{y}_n$ (a vector), temperature, and so on as inputs and the output will be $\boldsymbol{y}_{n+1}$ (a vector)
  - This function needs to call the function f defined previously

Hints for Task 1

- Please do not forget that the there is a condition on the temperature profile (cannot go below 5 ^{o}C)
- To get the whole profile y_0 to $y_{f,}$ you need to call iteratively the Runge-Kutta function defined previously using ‘for’/’while’ loop

Task 2 – Optimize k_g

- The growth kinetic parameter k_g used in task 1 is not the real one, estimate the optimum k_g based on the reference (measured) concentration profile given (different for each group) . Use the same cooling rate as Task 1.

  Note: The concentration profile file contain:

  column 1- time (in seconds)

  column 2- concentration

- Plot the predicted and reference concentration profiles vs time *before* and *after* optimization

Hints for Task 2

- Create a function to calculate the sum of squared error of concentration profile:

$$SSE = \sum_{t_i=0}^{t_f} \left[C_{predicted}\left(t_i\right) - C_{reference}\left(t_i\right)\right]^2$$

- Find the optimum k_g by iteration while calculating the SSE for each k_g. The optimum k_g is the one giving minimum SSE. For the iteration, use:

$$k_g = \exp(-7.5),\ \exp(-7.6),\ \exp(-7.7), \ldots, \exp(-16.0)$$

- Alternatively, you can use the built-in function '*fminsearch*' in Matlab (type '*help fminsearch*' in Matlab for instructions on how to use it)

Task 3 – Optimize cooling rate to satisfy a desired condition

- After we obtain the optimum kg, we're now confident that the model can mimic the real process. Then, we can optimize the cooling rate to achieve a desired condition. The desired condition is:
  - Maximize the final volume weighted mean size (μ_{43}) while having final yield at least 75%. State the optimum cooling rate, weighted mean size and final yield.
- The volume weighted mean size (μ_{43}) and yield expressions are given by:

$$\mu_{43}(t) = \frac{\mu_4(t)}{\mu_3(t)}$$

$$yield(t) = \frac{C_0 - C(t)}{C_0} \times 100\%$$

- Due to cooling capacity limitations of the crystallizer, our cooling rate is limited between 1 to 30 ^{o}C/hr

Task 3 – Optimize cooling rate to satisfy desired conditions

- For the above cooling rate range (with increment of 0.5 ^{o}C/hr), plot the profiles vs time for the following: (Each figure contains all profiles w.r.t all cooling rates):
  - Temperature profile
  - Concentration and saturation concentration
  - Volume weighted mean size
  - Yield

  Note: In the Figures, indicate the direction with respect to increasing cooling rate
- What are the optimal volume weighted mean size and yield?
- What happen if you operate with a cooling rate faster and lower than the optimal one? Please explain the reason

Task 3 – Optimize cooling rate to satisfy desired conditions

- If completion of crystallization at $t=t_f$ is indicated by supersaturation or $(C-C_{sat}) < 5x10^{-4}$, what will be the minimum cooling rate to satisfy this requirement? What happen if this cooling rate was larger than the optimum cooling rate you found previously?

Hints for Task 3

- Use similar iteration approach as Task 2.

  For the iteration, use cooling rate = 1 ^{o}C/hr, 1.5 ^{o}C/hr, ..., 30 ^{o}C/hr

Submission

- This project constitute 25% of the final grade
- Matlab codes
- Report (maximum 4 pages)

} Upload to Dropbox by 2 April (Sunday)

- Presentation slides (10 minutes) Upload to X-site by 3 April 24:00

Grading

- Matlab codes (~50%)
- Report (~25%)
- Presentation: content, clarity, skills, Q&A (~25%)
- Peer review

Notes

1. Please observe the maximum 4 pages for report
2. Each group will be given only 10 minutes for presentation on the results and approaches taken
3. You can consult me (28th March) on your approaches, but not debugging your Matlab codes
4. Ask all questions during consultation next week