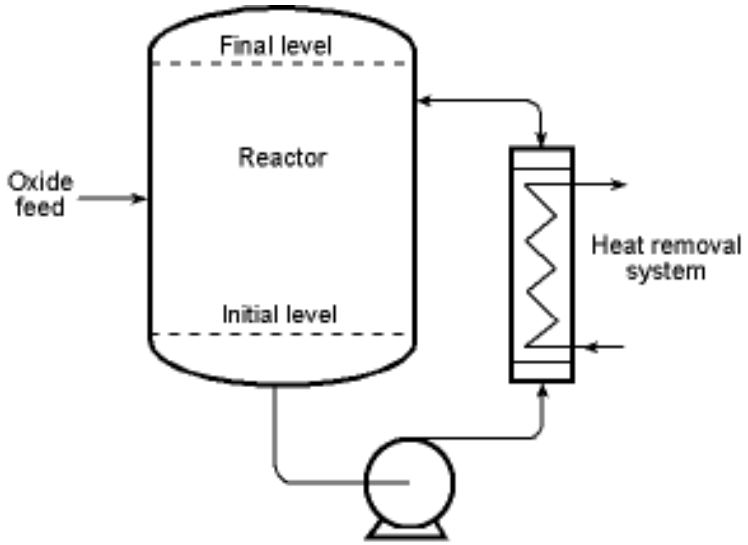
# SAFETY ANALYSIS WITH MODEL-BASED DYNAMIC SIMULATION ON MOBILE DEVICES

**Mordechai Shacham**
**and**
**Michael Elly**
**Ben Gurion University of the Negev**
**Beer-Sheva, Israel**

**Michael B. Cutlip**
**University of Connecticut**
**Storrs, CT**

# Example 1 – Temp. Runaway in a Polymerization Reaction




Consider the manufacture of a polyol lubricant by step-wise condensation of ***propylene oxide*** with ***butanol***:

$C_4H_9OH + (n+1)\ C_3H_6O \rightarrow$ heat + $C_4H_9(OC_3H_6)nOCH_2CHOHCH_3$

- The catalyzed alcohol is initially charged into the reactor, up to the "initial" level.
- The oxide is fed into the reactor at a constant rate, until the batch is ready and the reactor is full.
- Excess heat of the reaction is removed via an external heat removal system.

## Example 1 – Temp. Runaway in a Polymerization Reaction*

- Economical considerations dictate that the reaction should be completed at the ***highest possible rate***.
- The reaction rate is a function of the ***temperature, catalyst concentration*** and liquid phase oxide concentration (which is function of the ***pressure***).
- The limits on the reactor temperature and catalyst concentration are set by considerations of thermal degradation and purification difficulties.
- To maximize the reaction rate, ***the pressure must be kept as high as possible*** for the entire duration of the batch.

*Shacham et al. Computers & Chemical Engineering, 24, 415-421 (2000)

# The Motivation

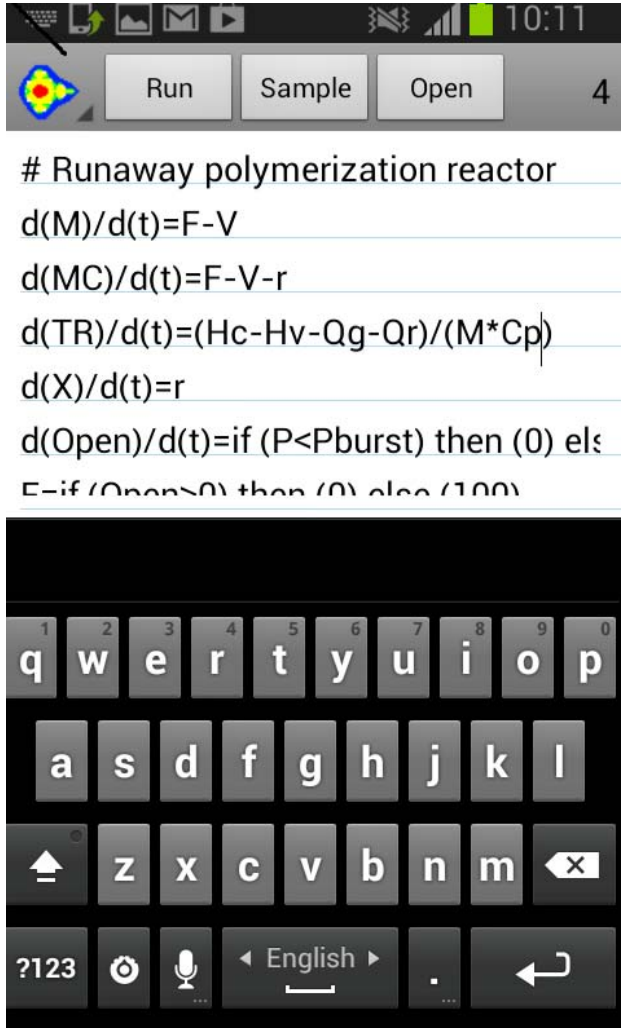
- The operator calls you in the middle of the night and tells you that there was a cooling failure of 10 minutes duration after about 700 minutes (11 h 40 min) operation of the reactor and asks you for instructions.

- You need to simulate the operation of the reactor taking into account the 10 minutes cooling failure to determine whether temperature runaway can be expected.

- In case the temperature runaway is imminent you need to try changes in the operating conditions that may prevent the temperature runaway

- You have only your smart-phone available.

# The PolyMathLite App*





*PolyMathLite is an Android app produced by PolyMath Software. (http://www.polymathlite.com)

## Example 1 –Typical Student Assignment

a. Simulate the ***normal operation*** of the semi-batch reactor ( ***Cooling recirculation flow rate***, $F_c$ = 3300 kg/min).

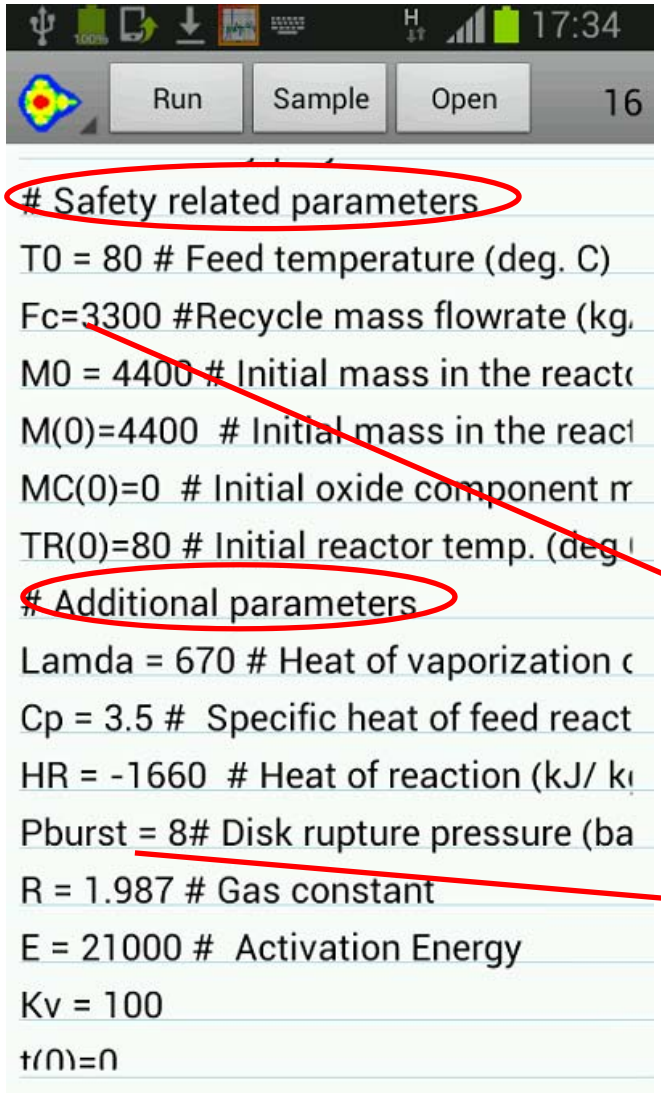
b. Check the effects of the ***reduction of*** $F_c$ by reducing it to $F_c$ = 2500 kg/min.

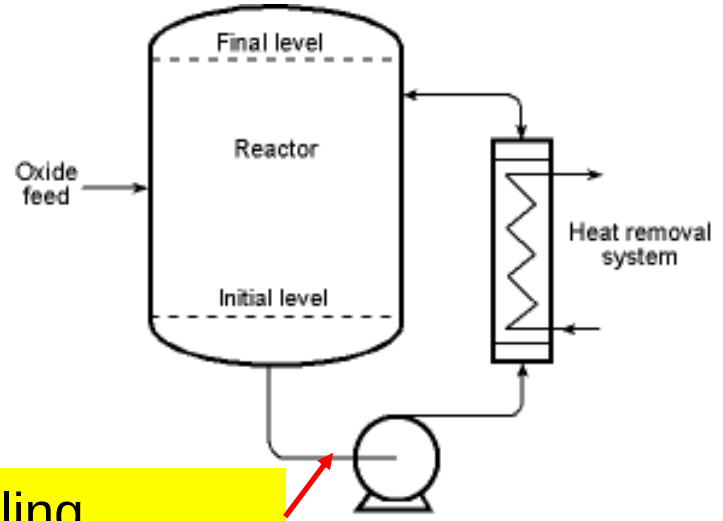
c. Check the effects of ***cooling failure*** with 10 minutes duration. Introduce such a failure 12 hrs after the startup.

d. Rework (c), but **increase** $F_c$ to 5000 kg/min ***after the failure.***

e. Check the effects of the ***failure of the bursting disk*** to open at the specified rupture pressure (= 8 atm).

# PolyMathLite Model of the Polymerization Reactor




Normal cooling recirculation flow-rate

Highest pressure allowed

# PolyMathLite Model of the Polymerization Reactor – Equation's Section

```
# Equations
d(M)/d(t) = F-V # Total mass in the reactor (kg)
d(MC)/d(t) = F-V-r # Oxide component mass in the reactor (kg)
d(TR)/d(t) =(Hc-Hv-Qg-Qr )/(M*Cp # Temperature in the reactor (deg C)
d(X)/d(t) = r # The mass of oxide reacted (kg)
d(Open)/d(t) = if (P<Pburst) then (0) else (0.001) # Status of the bursting disk: 0 closed, >0 open
F = if (Open>0) then (0) else (100) # Oxide feed rate (kg/min)
V = if ((P<=1) or (Open==0)) then (0) else (V1) # Vapor discharge rate (kg/min)
V1 = if (P<1.9) then (Vsubs) else (Vs) # Vapor discharge rate (kg/min)
Vs=0.85*Kv*P/sqrt(TR+273) # Sonic vapor discharge rate (kg/min)
Vsubs=Kv*P/sqrt((TR+273))*sqrt(1+1/P^2) # Sub-sonic - vapor discharge rate (kg/min)
r = k*MC # Reaction rate (kg oxide/min)
Hc= F*Cp*(T0-TR) # Feed enthalpy change (kJ/min)
Hv=V*Lamda # Latent heat of vapor discharge (kJ/min)
Qg=r*HR # Heat of reaction (kJ/min)
Qr=Fc*Cp*(TR-T0) # Heat removal rate ( kJ/min)
P = if (P1<1) then (1) else (P1) # Oxide vapor press
P1 = (exp(-3430/(TR+273)+11.7)+1.45e-3*MW)*C #
k = 9e9*exp(-E/(R*(TR+273))) # Reaction rate coeffi
C = MC/M # Oxide conncentration (kg/kg)
MW = (M0+X)/(M0/74) # Molecular weight of the polymer (kg/mol)
```

If the bursting disk is open the oxide feed stops and the vapor discharged

The model contains complete documentation, including the description of the variables and their units.

# Polymerization Reactor Results – Normal Operating Conditions

| | Variable | Initial value | Final value | Minimal value | Maximal value |
|---|---|---|---|---|---|
| 1 | C | 0 | 0.1578 | 0 | 0.744704 |
| 6 | Hc | 0 | -3955.89 | -1.11E+04 | 0 |
| 9 | k | 0.000895 | 0.002265 | 0.000895 | 0.010554 |
| 12 | M | 4400 | 2.04E+05 | 4400 | 2.04E+05 |
| 14 | MC | 0 | 3.23E+04 | 0 | 3.99E+04 |
| 15 | MW | 74 | 2895.18 | 74 | 2895.18 |
| 16 | Open | 0 | 0 | 0 | 0 |
| 17 | P | 1 | 2.21255 | 1 | 6.35029 |
| 18 | P1 | 0 | 2.21255 | 0 | 6.35029 |
| 20 | Qg | 0 | -1.21E+05 | -3.98E+05 | 0 |
| 21 | Qr | 0 | 1.31E+05 | 0 | 3.66E+05 |
| 22 | r | 0 | 73.0491 | 0 | 239.664 |
| 23 | R | 1.987 | 1.987 | 1.987 | 1.987 |
| 24 | t | 0 | 2000 | 0 | 2000 |
| 25 | T0 | 80 | 80 | 80 | 80 |
| 26 | TR | 80 | 91.3025 | 80 | 111.897 |
| 27 | V | 0 | 0 | 0 | 0 |
| 28 | V1 | 7.5271 | 9.8533 | 7.5271 | 28.4331 |
| 29 | Vs | 4.52409 | 9.8533 | 4.52409 | 28.4331 |
| 30 | Vsubs | 7.5271 | 12.7211 | 7.5271 | 33.863 |
| 31 | X | 0 | 1.68E+05 | 0 | 1.68E+05 |

Molecular weight of the polymer (kg/mol)

Maximal pressure (bar)

Maximal temp. (°C) in the reactor

# Reactor's Operation with Cooling Media Failure

## Fc=if (*t*<700) then (3300) else (if (*t*>710) then (3300) else (0))

| | Variable | Initial value | Final value | Minimal value | Maximal value |
|---|---|---|---|---|---|
| 1 | C | 0 | 7.82E-25 | 0 | 0.744704 |
| 6 | Hc | 0 | 0 | -5663.84 | 0 |
| 9 | k | 0.000895 | 0.000895 | 0.000895 | 4.08745 |
| 12 | M | 4400 | 7.55E+04 | 4400 | 7.59E+04 |
| 14 | MC | 0 | 5.90E-20 | 0 | 3.81E+04 |
| 15 | MW | 74 | 1269.51 | 74 | 1269.51 |
| 16 | Open | 0 | 0.01324 | 0 | 0.01324 |
| 17 | P | 1 | 1 | 1 | 8.30799 |
| 18 | P1 | 0 | 7.12E-24 | 0 | 8.30799 |
| 20 | Qg | 0 | -8.76E-20 | -7.85E+05 | 0 |
| 21 | Qr | 0 | 1.64E-10 | 0 | 1.60E+06 |
| 22 | r | 0 | 5.28E-23 | 0 | 472.828 |
| 23 | R | 1.987 | 1.987 | 1.987 | 1.987 |
| 24 | t | 0 | 2000 | 0 | 2000 |
| 26 | TR | 80 | 80 | 80 | 215.563 |
| 28 | V1 | 7.5271 | 7.5271 | 6.38044 | 35.8127 |
| 29 | Vs | 4.52409 | 4.52409 | 3.8349 | 35.8127 |
| 30 | Vsubs | 7.5271 | 7.5271 | 6.38044 | 42.4368 |
| 31 | X | 0 | 7.11E+04 | 0 | 7.11E+04 |

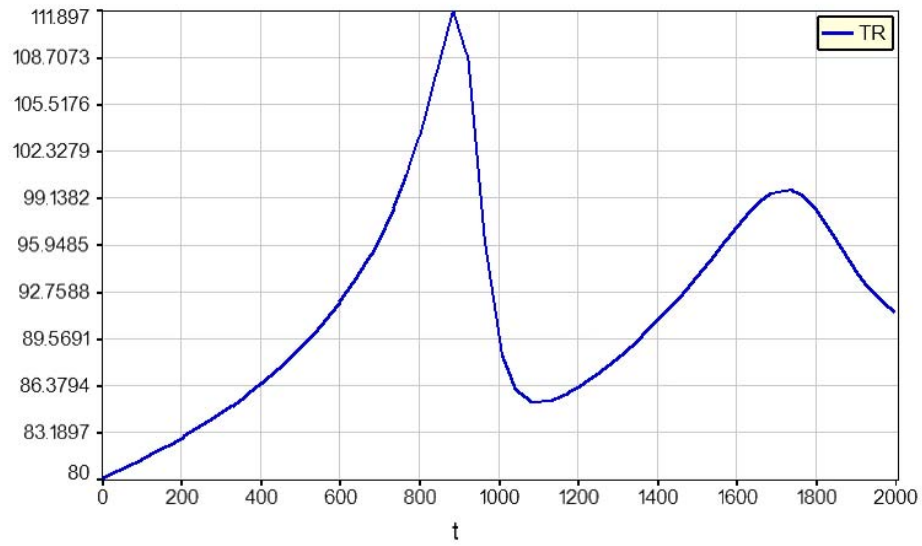
Molecular weight of the polymer (kg/mol)

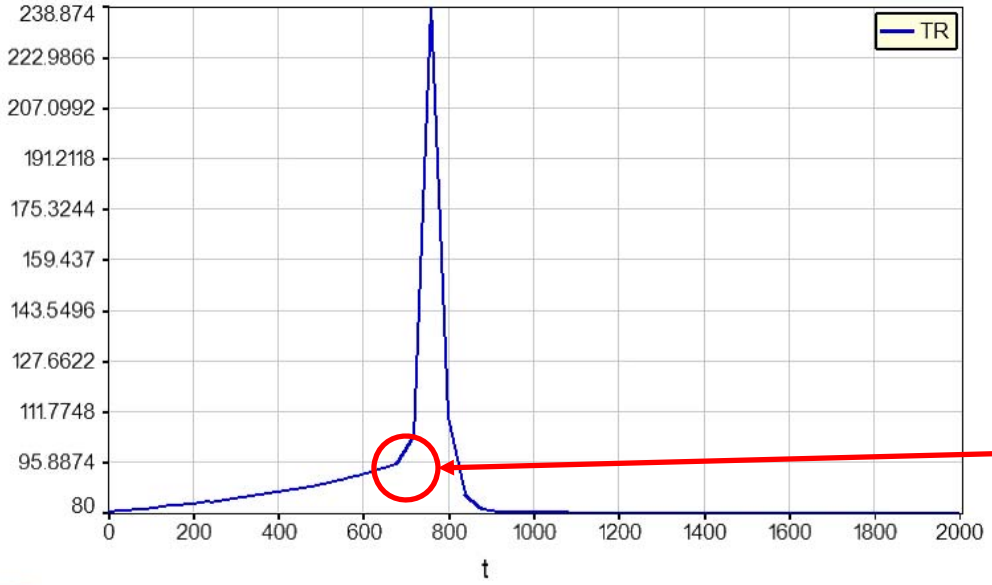
Maximal pressure (bar)

Maximal temp. (° C) in the reactor

# Reactor's Temperature - Normal Operation and Temperature Runaway

Normal operation



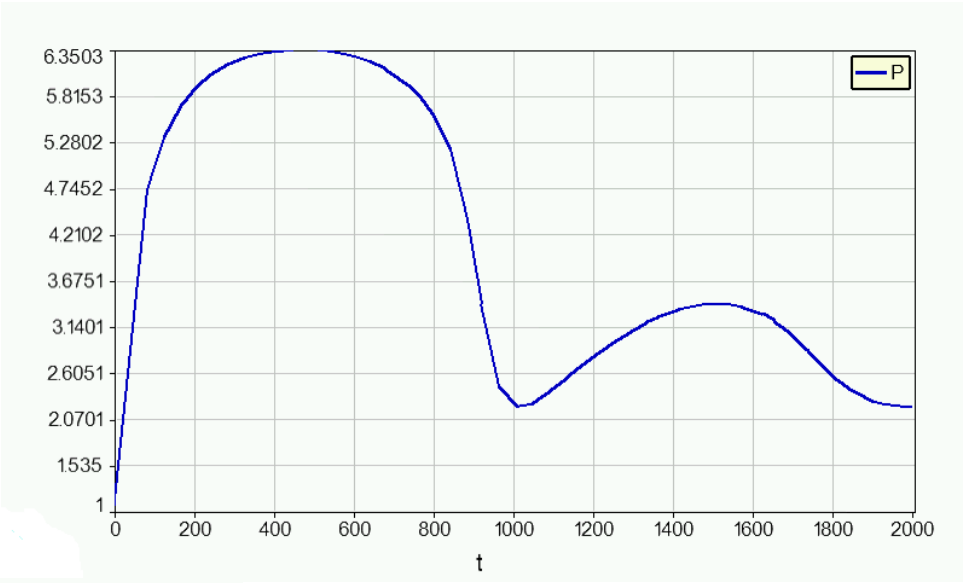



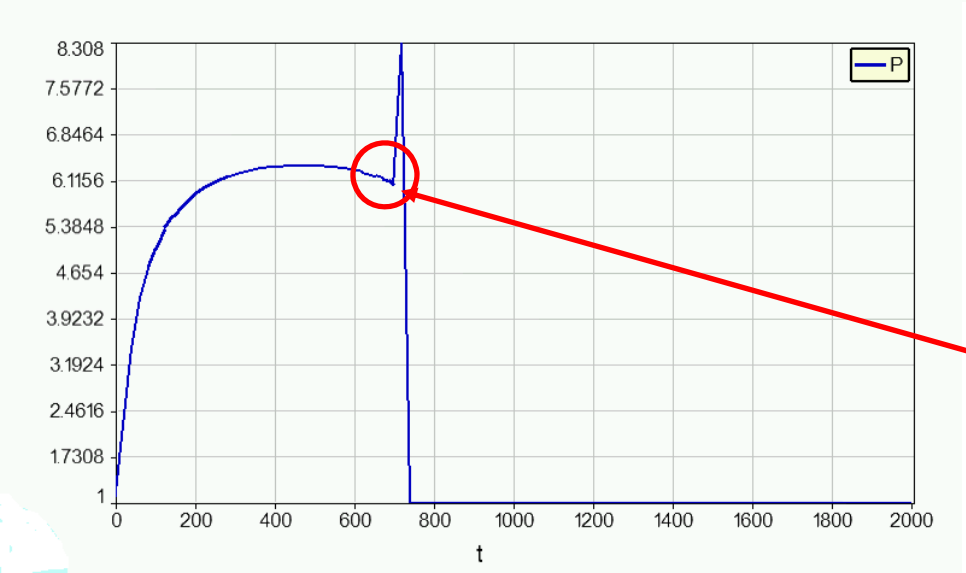

Temperature runaway

Cooling failure
for 10 minutes

# Reactor's Pressure - Normal Operation and Temperature Runaway

Normal operation




Temperature runaway

Cooling failure
for 10 minutes

## Results of the Safety Analysis for Example 1

a. Simulate the normal operation of the semi-batch reactor

$T_{max}$ = 111.8 °C; $P_{max}$ = 6.35 bar and $MW_{final}$ = 2895 kg/mol;

b. Check the effects of the reduction of the cooling re-circulation mass flow rate to $F_c$ =2500 kg/min

$T_{max}$ = 303 °C; $P_{max}$ = 8.15 bar and $MW_{final}$ = 893 kg/mol;

c. Check the effects of cooling failure with 10 minutes duration

$T_{max}$ = 215.6 °C; $P_{max}$ = 8.3 bar and $MW_{final}$ = 1269 kg/mol;

d. Rework (c), but increase $F_c$ to 5000 kg/min after the failure.

$T_{max}$ = 105.6 °C; $P_{max}$ = 6.9 bar and $MW_{final}$ = 2565 kg/mol;

e. Check the effects of the failure of the bursting disk to open

The model cannot handle properly such a scenario

# Export of the Polymerization Reactor Model to MATLAB

09:56

Polymath Report

**Matlab formatted problem**

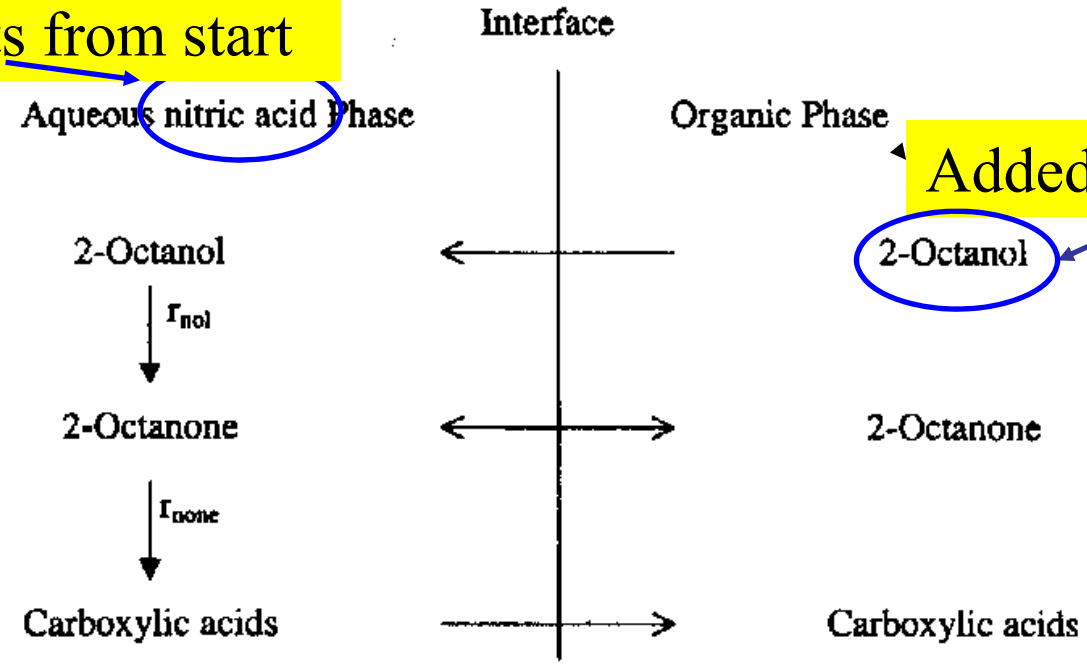
Create m file called PolyOde.m and paste the following text into it.

```
% Polymath ODE problem conversion to Matlab
% DEQ
function PolyOde
  tspan = [0 2000]; % Range for the independent variable
  y0 = [4400; 0; 80; 0; 0]; % Initial values for the dependent varia
  [t,y]=ode45(@ODEfun,tspan, y0);
  plot (t,y);
  xlabel('t');
  legend('M','MC','TR','X','Open');
  fprintf('M = %16.6f \n',y(length(y),1));
  fprintf('MC = %16.6f \n',y(length(y),2));
  fprintf('TR = %16.6f \n',y(length(y),3));
  fprintf('X = %16.6f \n',y(length(y),4));
  fprintf('Open = %16.6f \n',y(length(y),5));
end

function dYfuncvecdt = ODEfun(t,Yfuncvec)
  M = Yfuncvec(1);
  MC = Yfuncvec(2);
  TR = Yfuncvec(3);
  X = Yfuncvec(4);
  Open = Yfuncvec(5);
  if (Open > 0)
    F = 0;
  else
    F = 100;
  end
  C = MC / M;
  M0 = 4400;
  MW = (M0 + X) / (M0 / 74);
  Kv = 100;
  R = 1.987;
  Cp = 3.5;
  P1 = (exp(-3430 / (TR + 273) + 11.7) + 0.00145 * MW) * C;
```

```
  E = 21000;
  if (t < 700)
    Fc = 3300;
  else
    if (t > 705)
      Fc = 3300;
    else
      Fc = 0;
    end
  end
  if (P1 < 1)
    P = 1;
  else
    P = P1;
  end
  Vs = 0.85 * Kv * P / sqrt(TR + 273);
  k = 9000000000 * exp(0 - (E / (R * (TR + 273))));
  Vsubs = Kv * P / sqrt(TR + 273) * sqrt(1 + 1 / (P ^ 2));
  if (P < 1.9)
    V1 = Vsubs;
  else
    V1 = Vs;
  end
  T0 = 80;
  Lamda = 670;
  Hc = F * Cp * (T0 - TR);
  HR = -1660;
  Qr = Fc * Cp * (TR - T0);
  Pburst = 8;
  r = k * MC;
  Qg = r * HR;
  if (P <= 1 | (Open == 0))
    V = 0;
  else
    V = V1;
  end
  Hv = V * Lamda;
  dMdt = F - V;
  dMCdt = F - V - r;
  dTRdt = (Hc - Hv - Qg - Qr) / (M * Cp);
  dXdt = r;
  if (P < Pburst)
    dOpendt = 0;
  else
    dOpendt = 0.001;
  end
  dYfuncvecdt = [dMdt; dMCdt; dTRdt; dXdt; dOpendt];
end
```

# Example 2 -Nitric Acid oxidation of 2-octanol in a Semi-Batch Reactor




$$A + B \xrightarrow{r_1} P + 2B$$

$$P + B \xrightarrow{r_2} X$$

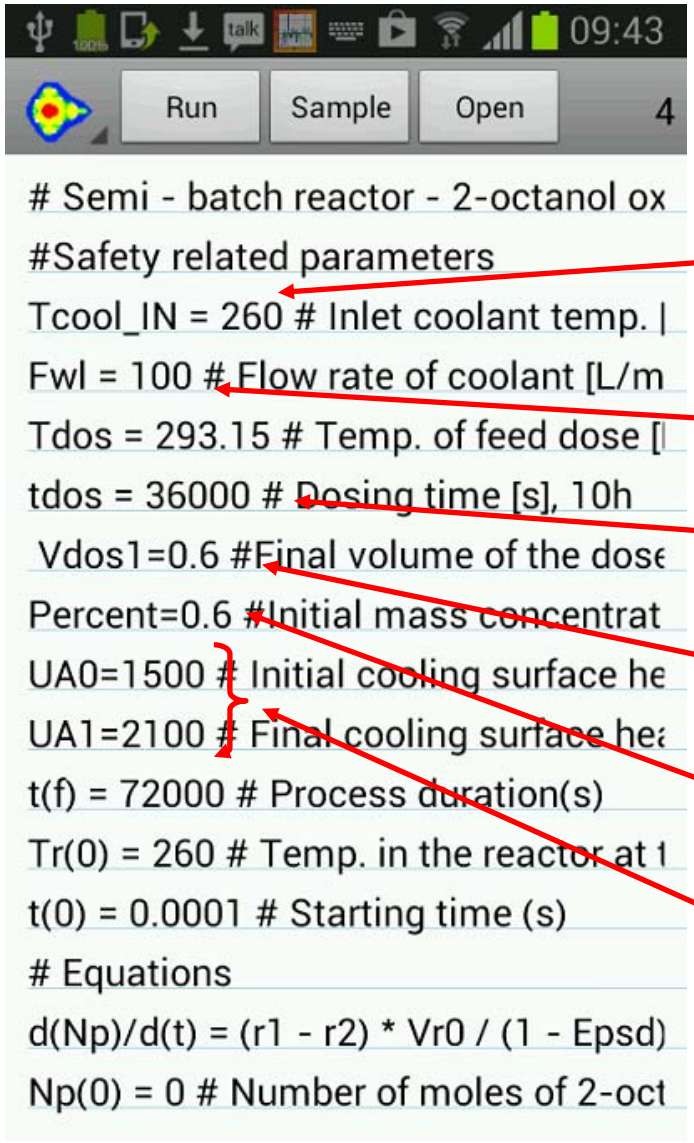
*A* - 2- octanol, *P* - 2-octanone , *X* – undesired oxidation products B - nitrosonium ion, causes an autocatalytic behavior

## Example 2 -Nitric Acid oxidation of 2-octanol in a Semi-Batch Reactor (van Woezik and Westerterp*)

- In the semi-batch reactor aqueous nitric acid is initially present, and 2- octanol (A) is added at a constant rate until a desired molar ratio of the reactants is reached.
- The 2-octanol reacts to form 2-octanone and carboxylic acid.
- Under normal operating conditions, the ***temperature*** in the reactor ***does not exceed the limit of ~ 0*** °C.
- If the ***temperature at any time exceeds ~ 5 °C, runaway*** conditions develop, which may lead to a maximal temperature of ***over 200 °C***, and conversion of all of the ***2-octanone to carboxylic acid.***

**Chemical Engineering and Processing*, *39*, 521(2000) and *41*, 59(2001).

# Example 2 – Important Safety Related Parameters




## Potential Causes of Temperature Runaway

- Increase of coolant inlet temp.
- Reduction of coolant flow rate
- Shortening of dosing time
- Increase of dose volume
- Increase of nitric acid concentration
- Reduction of effective heat transfer rate

# PolyMAthLite Model of the of the 2-octanol Oxidation Reactor– Mole Balance Equations

```
# Equations
d(Np)/d(t) = (r1 - r2) * Vr0 / (1 - Epsd) # Mole balance for 2-octanone (P)
Np(0) = 0 # Number of moles of 2-octanone (P) at t = t0
d(Nx)/d(t) = r2 * Vr0 / (1 - Epsd) # Mole balance for carboxylic acids (X)
Nx(0) = 0 # Number of moles of carboxylic acids (X) at t = t0
r1 = k1 * CaOrg * CbAq * (1 - Epsd) # Reaction rate of a and b to p[kmol/m3/s]
r2 = k2 * CpOrg * CbAq * (1 - Epsd) # Reaction rate of p and b to x[kmol/m3/s]
Vr0 = 1.5 # Initial volume in a reactor [m3]
Epsd = Vdos1 / (Vdos1 + Vr0) # Volume fraction of dispersed phase
k1 = maA1 * exp(-E1perR / Tr - m1 * H) # Specific reaction rate 1
k2 = mpA2 * exp(-E2perR / Tr - m2 * H) # Specific reaction rate 2
CaOrg = (Theta * NaF - Np - Nx) / (Vdos1 * Theta) # Concentr of a in org phase [kmole/m3]
CpOrg = Np / (Vdos1 * Theta) # Concentr. of (P) in org phase [kmol/m3]
CbAq = (Np + Y * NaF) / Vr0 # Concentr. of (B) in aq. phase [kmole/m3]
maA1 = 10 ^ 5 # Pre-exponential factor reaction 1 [m3/kmol/s]
mpA2 = 10 ^ 10 # Pre-exponential factor reaction 2[m3/kmol/s]
E1perR = 11300 # Activation temperature reaction 1 [K]
E2perR = 12000 # Activation tempetature reaction 2 [K]
m1 = 6.6 # Hammett's reaction rate coeff. reaction 1
m2 = 2.2 # Hammett's reaction rate coeff. reaction 2
H = -0.6221 - 3.7214 * wt - 1.5714 * wt ^ 2 # Hammett's acidity function
Theta = If (t <= tdos) Then (t / tdos) Else (1) # Dimensionless time up to t=tdos
NaF = Vdos1 * RhoOctan / MwOctan # Total amount of 2-octanol (a) fed [kmol]
Y = 0.035 # Initial concentr. of nitrosonium ion Y=Nb0/NaF
wt = Nn * Mw / (Vr0 * RhoAcid) # Mass concentr. of nitric acid sol [%/100%]
RhoOctan = 820.7 # Density of 2-octanol [kg/m3]
MwOctan = 130.23 # Molar mass of 2-octanol [kg/kmol]
Nn = CnAq * Vr0 # Number of moles of HNO3 [kmol]
Mw = 63 # Molar mass of HNO3 [kg/kmol]
RhoAcid = 1500 # Density of pure nitric acid [kg/m3]
CnAq = (NnO - Y * NaF - Np - 2 * Nx) / Vr0 # Concentr. of HNO3 in the aq. phase [kmol/m3]
```

# PolyMAthLite Model of the of the 2-octanol Oxidation Reactor– Energy Balance Equations

```
d(Tr)/d(t) = (Qr + Qdos + Qcool) / Gamma # Reactor energy balance (Tr in K)
Qr = Qnol + Qnone # Sum of the heat of reaction the reactions [W)
Qdos = Phi * RhoCPdos * (Tdos - Tr) # Heat input due to reactant addition [W]
Qcool = UAcool * (Tcool - Tr) # Heat removed by the cooling jacket [W]
Gamma = Gamma0 + RhoCPdos * Phi * t # Total heat capacity of the system [J/K]
Qnol = r1 * Vr0 * Hnol / (1 - Epsd) # Heat of reaction, 1 [W]
Qnone = r2 * Vr0 * Hnone / (1 - Epsd) # Heat of reaction, 2 [W]
Phi = Vdos1 / tdos # Volumetric flow rate of the feed [m3/s]
RhoCPdos = 2 * 10 ^ 6 # Heat capacity of dose [J/m3/K]
UAcool = UA0 + (UA1 - UA0) * Theta # Cooling surface heat transfer coefficient [W/K]
Gamma0 = 5.4 * 10 ^ 6 # Initial heat capacity of the system [J/K]
Hnol = 160 * 10 ^ 6 # Specific heat of reaction 1 [J/kmol]
Hnone = 520 * 10 ^ 6 # Specific heat of reaction 2 (J/kmol)
Fw = Fwl / 60 * 10 ^ (-3) # Flow rate of coolant [m3/s]
d(Tcool)/d(t) = (Fw * (Tcool_IN - Tcool) - Qcool / (RhoCoolant * CpCoolant)) / Vj # Jacket energy balance (T in K)
Tcool(0) = 273.15 # Coolant exit temp. at t = t0 (K)
RhoCoolant = 1000 # The density of coolant [kg/m3]
CpCoolant = 4183 # Heat capacity of coolant [J/kg/K]
Vj = 1.5 # Volume of the jacket [m3]
```

# 2-octanol Oxidation Reactor– Simulation Results

## Normal operating conditions

| | Variable | Initial value | Final value | Maximal value |
|---|---|---|---|---|
| Cooling Temp. in (K) | Tcool_IN | 260 | 260 | 260 |
| Cooling Temp. out (K) | Tcool | 273.15 | 260.38 | 273.15 |
| Reactors temp. (K) | Tr | 260 | 261.64 | 274.57 |
| 2-octanone (mol) | Np | 0 | 3.1224 | 3.1224 |
| Carboxylic acid (mol) | Nx | 0 | 0.2521 | 0.2521 |

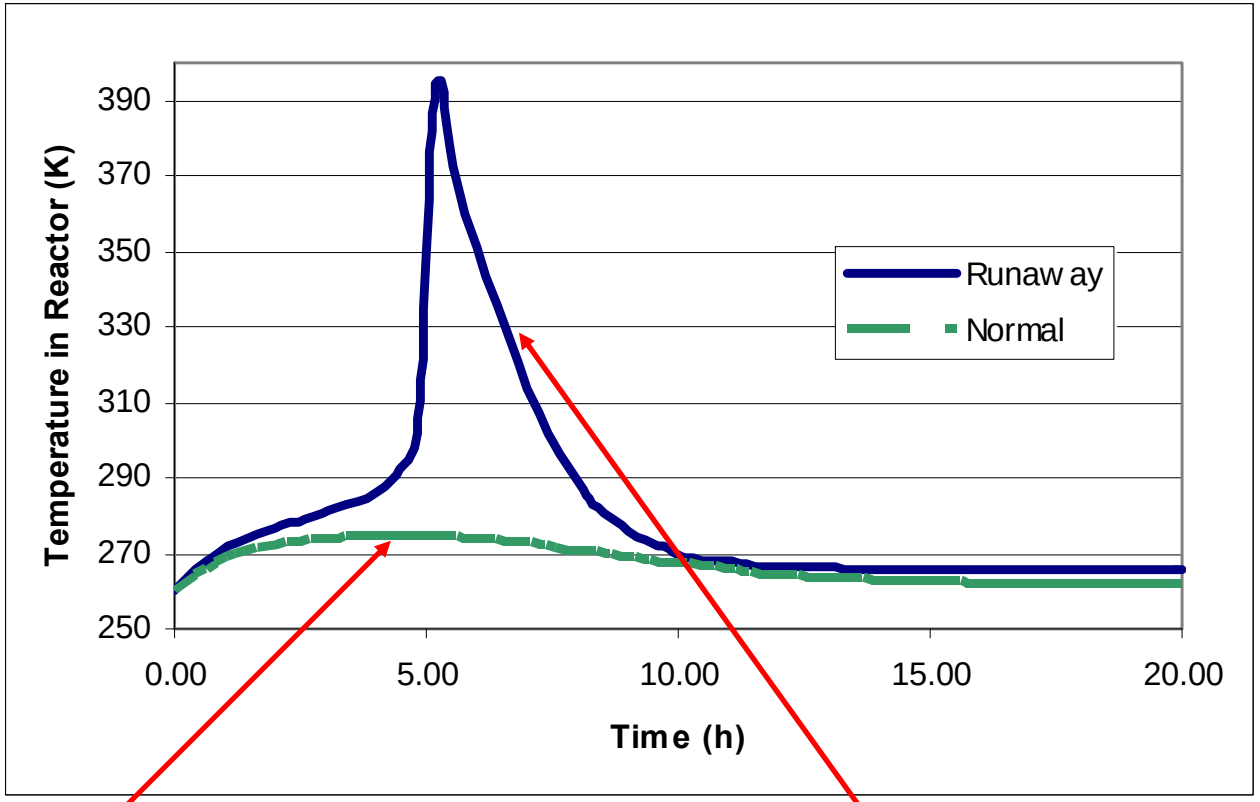
Desired product

Temperature runaway

Maximal temp. (K)

| | Variable | Initial value | Final value | Maximal value |
|---|---|---|---|---|
| Cooling Temp. in (K) | Tcool_IN | 265 | 265 | 265 |
| Cooling Temp. out (K) | Tcool | 273.15 | 265.16 | 288.12 |
| Reactors temp. (K) | Tr | 260 | 265.67 | 392.98 |
| 2-octanone (mol) | Np | 0 | 0.3353 | 1.3596 |
| Carboxylic acid (mol) | Nx | 0 | 2.4050 | 2.4050 |

# Reactor's Temperature - Normal Operation and Temperature Runaway




Normal operation

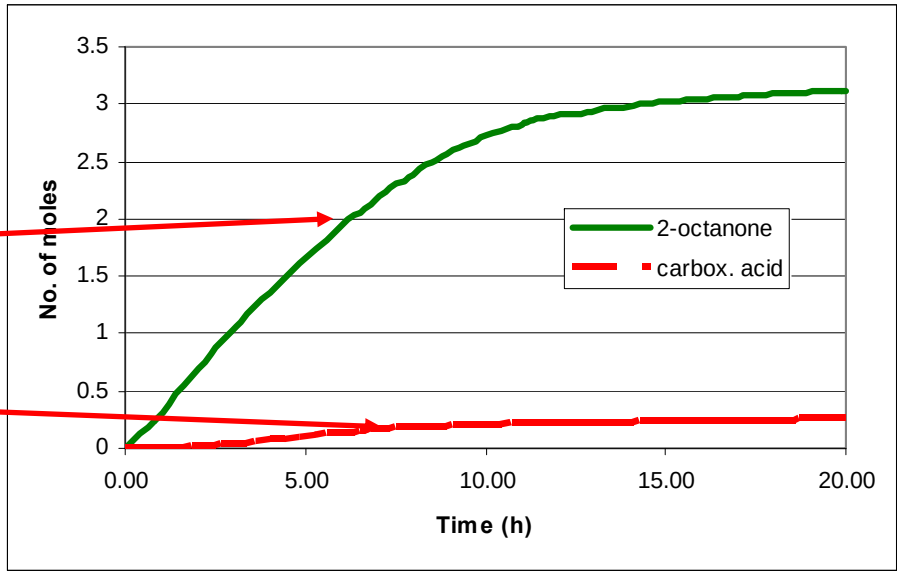
Temperature runaway

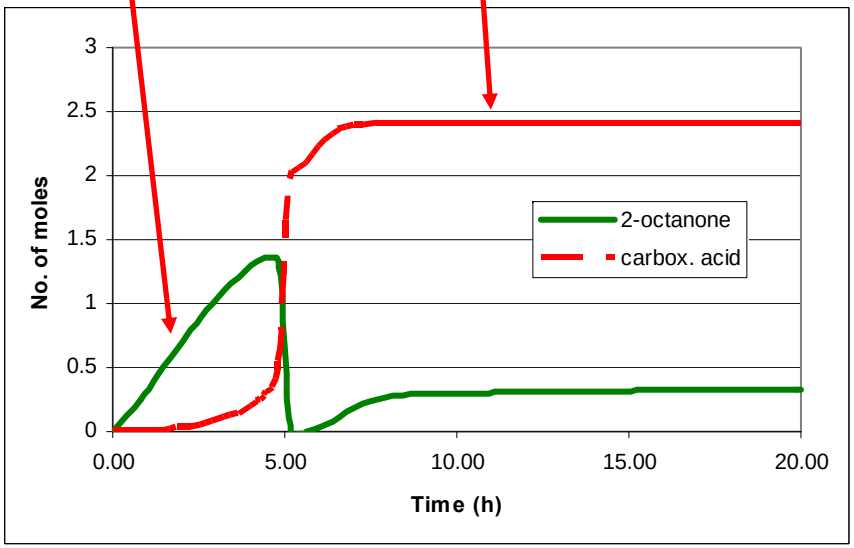
# Product Distribution - Normal Operation and Temperature Runaway

Normal operation

2-octanone – desired product

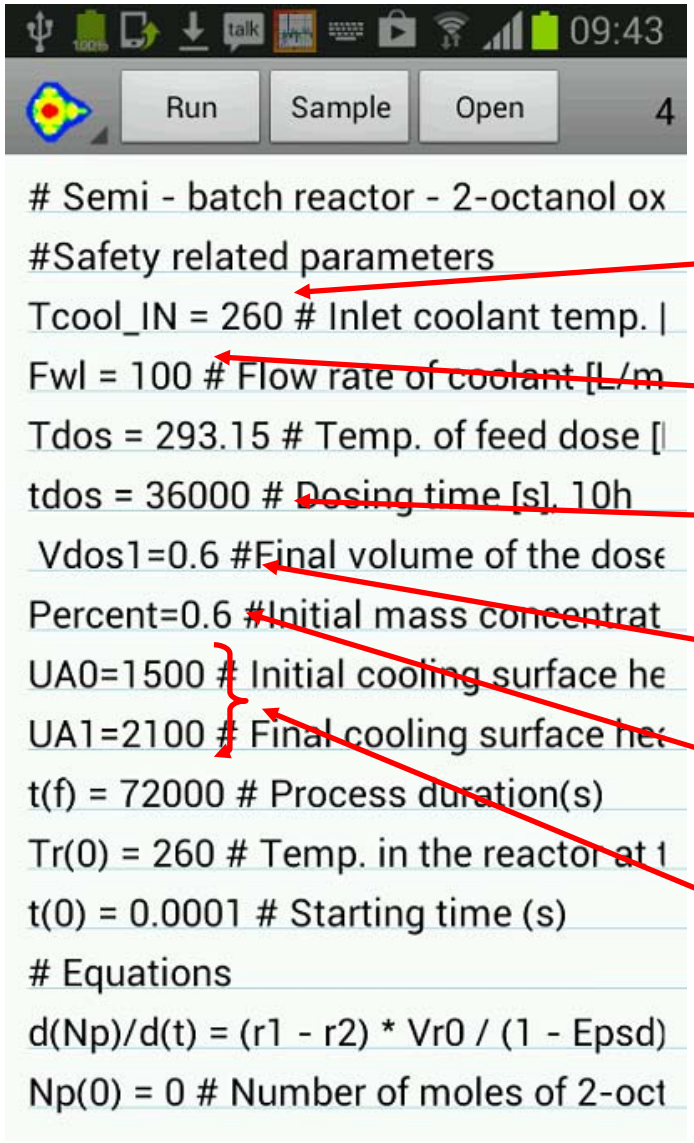
Carboxylic acid - undesired product




Temperature runaway

# Example 2 - Additional Causes of Temperature Runaway




Increase of coolant inlet temp. **≥ 266 K**

Reduction of coolant flow rate **≤ 12**

Shortening of dosing time **≤ 7.5 h**

Increase of dose volume **≥ 0.81 $m^3$**

Increase of nitric acid concentration **≥ 0.64**

Reduction of effective heat transfer rate **≤ 81%**

## Conclusions

- It has been shown that model-based dynamic simulation of processes enable students to ***learn*** safety related issues ***by discovering potential safety hazards and trying out methods for their prevention***.

- This method of instruction makes it possible to ***incorporate process safety into various chemical engineering courses***, such as process design, process simulation and reaction engineering.

- The availability of PolyMathLite (or other mobile device based mathematical modeling programs) can considerably extend the use of MDBS for safety analysis, ***beyond the classroom, eliminating the need to use large scale simulation programs.***