Effect of The Precipitation of Acid Soap and Alkanoic Acid Crystallites on The Bulk pH

Gergana Velikova, Sofia University, FMI
Ivan Georgiev, IMI-BAS/IICT-BAS
Milena Veneva, Sofia University, FMI

The 104th European Study Group with Industry
ESGI'104
Sofia, Bulgaria

September, 23 - 27, 2014

Plan of The Presentation:

The Problem

$$\text{Soap} = \text{fatty acids} + \text{base (NaOH/KOH)}$$

- K_a - fatty acid's dissociation constant;
- K_w - water's dissociation constant;
- Q_{mz} - rate constant of the soap production;
- K_{CO2} - used, because of the solubility of the CO_2 from the atmosphere.

Dissociation constants are type of rate constants.

The Problem

- System of polynomial equations in more than one variable: $F_j(x_1, x_2, \ldots, x_N) = b_j, j = \overline{1, N}$
- goal 1: fast algorithm for solving the system;
- goal 2: fast algorithm for finding the positive solution;
- goal 3: fitting the theoretically evaluated data for *pH* with the experimentally obtained one;
- goal 4: high precision of the solution.

The Problem

- C_H - concentration of the hydrogen cations;
- C_Z - concentration of the fatty acid anions;
- C_M - concentration of the metal cations;
- C_{MZ} - concentration of the soaps;
- C_{OH} - concentration of the hydroxide anions;
- C_{HCO3} - concentration of the hydrogencarbonate anions;
- C_a - concentration of the added salt (NaCl);
- C_b - concentration of the added base (NaOH);
- C_{HZ} - concentration of the undissociated fatty acid.

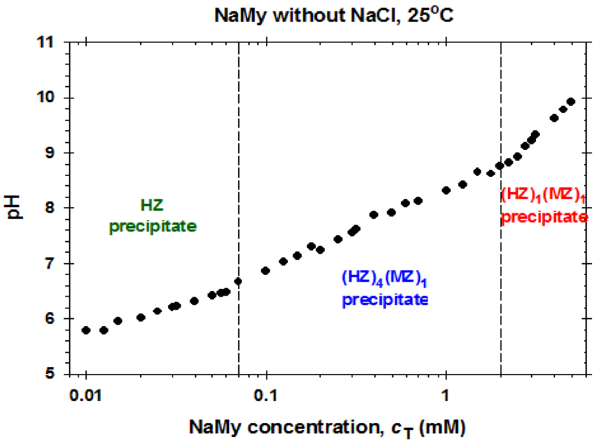
Mathematical Formulation of The Problem

- $C_H C_Z \ \gamma_{\pm}^2 = K_A \ C_{HZ}$
- $C_M C_Z \ \gamma_{\pm}^2 = Q_{MZ} \ C_{MZ}$
- $C_H C_{OH} \ \gamma_{\pm}^2 = K_W$
- $C_H C_{HCO3} \ \gamma_{\pm}^2 = K_{CO2}$
- $I = C_H + C_M = C_{OH} + C_{HCO3} + C_Z + C_A$
- $m_M = C_T + C_A + C_B - C_M - C_{MZ}$
- $m_Z = C_T - C_Z - C_{HZ} - C_{MZ}$

Mathematical Formulation of The Problem

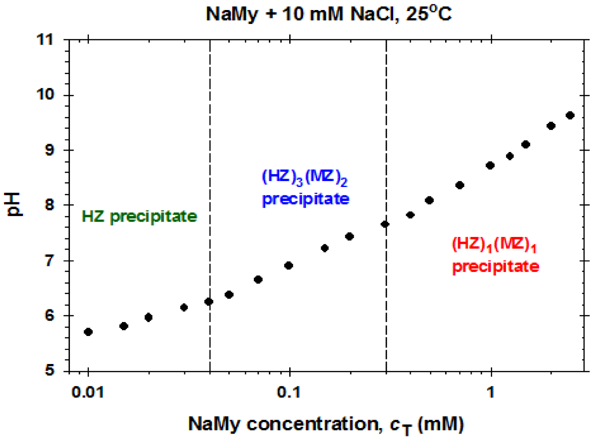
- $K_W = 6.81x10^{-15}$ M^2
- $K_A = 1.995x10^{-5}$ M
- $Q_{MZ} = 2.84$ M
- $\log_{10} \gamma_{\pm} = 0.055I - \frac{0.5115\sqrt{I}}{1 + 1.316\sqrt{I}}$ semi-empirical formula
- $pH = -\log_{10}(\gamma_{\pm} C_H)$

First Case - without NaCl



- $C_A = 0\,\mathrm{M}$ and $C_B = 0\,\mathrm{M}$

Second Case - with NaCl



- $C_A = 0.01\,\mathrm{M}$ and $C_B = 0\,\mathrm{M}$

First and Second Case - First Interval

- solution with fatty acid precipitates
- $C_{HZ} = S_{HZ} = 5.25x10^{-7}$ M
- $m_M = 0$

$\Rightarrow K_{CO2}$
$\Rightarrow$ comparison between the obtained K_{CO2} values in the two cases.

First and Second Case - Second Interval

- solution with precipitate of j:n acid soap
- $\dfrac{m_M}{n} = \dfrac{m_Z}{n+j}$
- $C_H^j C_M^n C_Z^{j+n} \gamma_{\pm}^{2j+2n} = K_{jn}, \quad \text{if } j = 4 \text{ and } n = 1$
- $C_H^j C_M^n C_Z^{j+n} \gamma_{\pm}^{2j+2n} = K_{jn}, \quad \text{if } j = 3 \text{ and } n = 2$

$\Rightarrow K_{41}$
$\Rightarrow K_{32}$

First and Second Case - Third Interval

- solution with precipitate of j:n acid soap
- $\dfrac{m_M}{n} = \dfrac{m_Z}{n+j}$
- $C_H^j C_M^n C_Z^{j+n} \gamma_\pm^{2j+2n} = K_{jn}, \quad \text{if } j = 1 \text{ and } n = 1$

$\Rightarrow K_{11}$

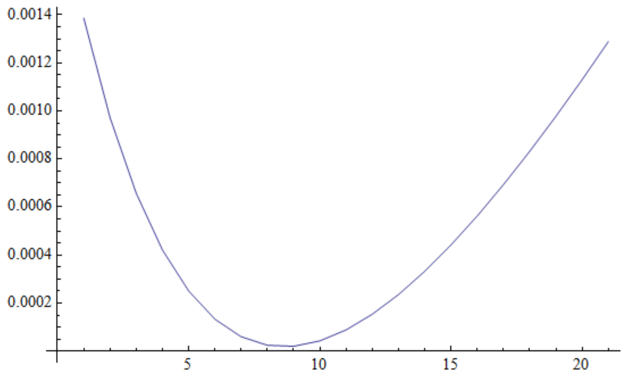
$\Rightarrow$ comparison between the obtained K_{11} values in the two cases.

Solution

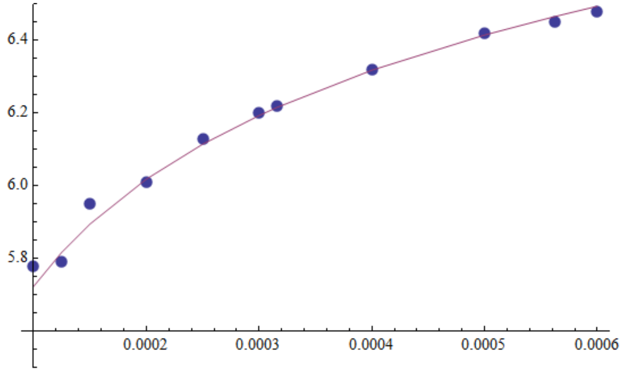
In order to fit the theoretically evaluated data with the experimentally obtained one, we minimize the following functional:

$$P(K_{CO2}) = \frac{1}{n}\sum_{k=1}^{n}\left[1 - \frac{pH_{th}(k)}{pH_{exp}(k)}\right]^2$$

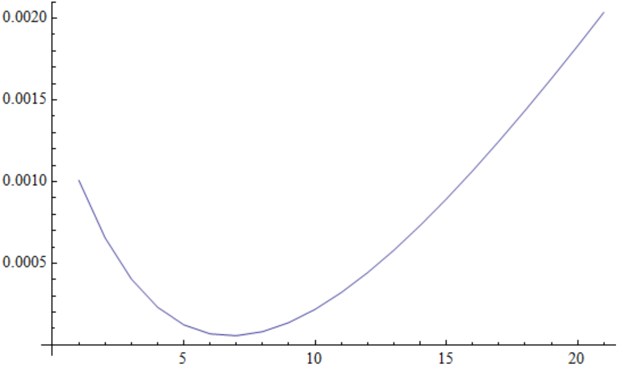
First Case (first interval) - values of $P(K_{CO2})$ using numerical variation of K_{CO2}



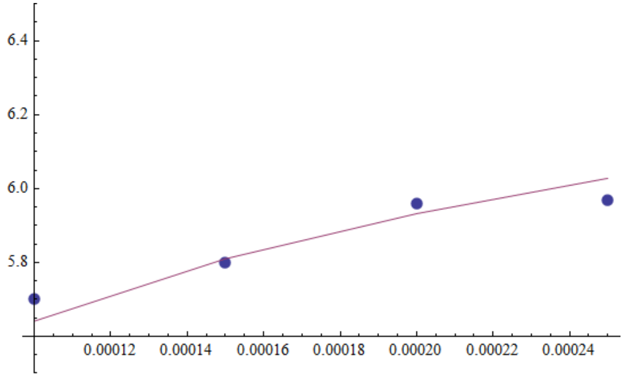
First Case (first interval) - fit of the theoretically evaluated data for *pH* with the experimentally obtained one ($K_{CO2} \approx 1.8x10^{-10}$)



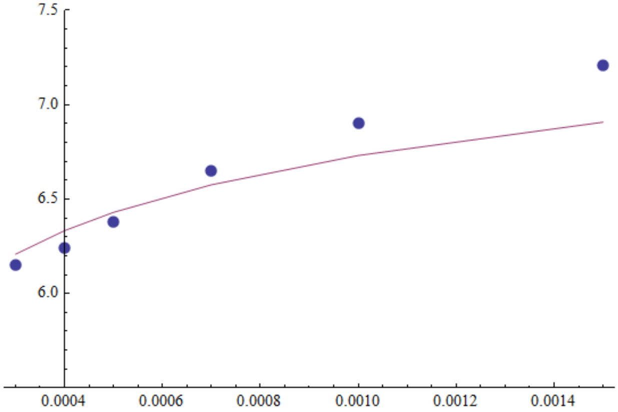
Second Case (first interval) - values of $P(K_{CO2})$ using numerical variation of K_{CO2}



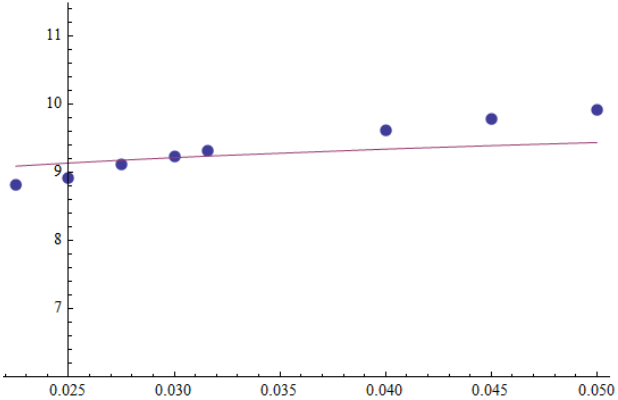
Second Case (first interval) - fit of the theoretically evaluated data for *pH* with the experimentally obtained one ($K_{CO2} \approx 2x10^{-10}$)



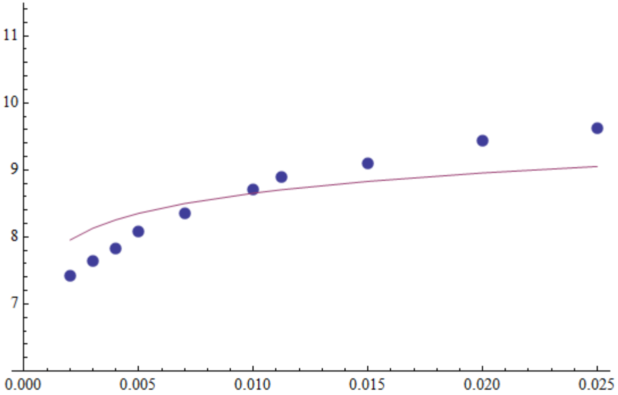
Second Case (second interval) - fit of the theoretically evaluated data for *pH* with the experimentally obtained one ($K_{32} \approx 1x10^{-51}$)



First Case (third interval) - fit of the theoretically evaluated data for *pH* with the experimentally obtained one ($K_{11} \approx 1x10^{-20}$)



Second Case (third interval) - fit of the theoretically evaluated data for *pH* with the experimentally obtained one ($K_{11} \approx 1x10^{-20}$)



Fast Algorithm for Finding The Positive Solution

Mathematical Problem

A system for which we know that:

- consists of $\leqslant 20$ polynomial equations;
- there is no estimation for the number of the solutions, because it depends on the type of the crystals;
- the solutions can be complex;
- only one solution with positive components exists (a hypothesis).

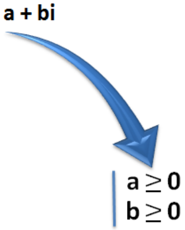
Goal

Example: system consisting of 16 equations with 16 variables. The solutions (obtained with *Mathematica*) are 9.

Goal: fast algorithm for finding the positive solution.

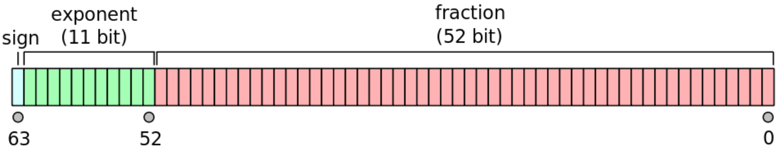
Methods for Finding The Positive Solution - first approach

Comparison with 0:



Complexity of the algorithm: $O(n * m)$, where:
n is the number of the solutions of the system,
m is the number of the components in each solution.

Methods for Finding The Positive Solution - second approach - introduction



- each double-precision floating-point number in the computer memory is $8B = 64 bits$ (according to the standard IEEE);
- 1 *bit* (the sign bit) is 0, if the number is ≥ 0 and 1, if it is negative.

$\Rightarrow$ 15 decimal digits in the decimal part of the mantis and the absolute value is between 10^{-308} and 10^{308}.

Methods for Finding The Positive Solution - second approach

Second approach: looking only the sign bit of each number. Then:

- complexity of the algorithm *comparison with 0* is: $O(l * n * m)$;
- complexity of the algorithm *bit comparison* is: $O(n * m)$,

where l is the number of the bits in the binary representation of the numbers, which we consider. In our case - 64.

In the worst case scenario, the modified algorithm works as fast as the first one. It depends on the optimizations that the processor makes.

C++/Fortran vs. *Matlab/Mathematica*

- *C++* and *Fortran* are compiled programming languages (the source code of the program is transformed into a machine code before the execution of the program);
- *Matlab* and *Mathematica* are interpreted programming languages (the programs are executed directly, which usually makes them slower because of the overhead of the processor).

$\Rightarrow$ *C++* and *Fortran* are better for scientific computations.

Implementation with *MATLAB* - time (s)

Bit Comparison	Comparison with 0
3.683144e-005	2.888495e-005
5.576608e-005	3.135687e-005
2.870890e-005	2.804986e-005
4.362872e-005	4.470864e-005
3.317848e-005	3.355881e-005

Here we can see that the algorithm *bit comparison* is slower. The reason is that the function (build-in), which *Matlab* uses for finding the sign bit, probably has the following implementation (with some optimizations): sign v = -(v $<$ 0);
$\Rightarrow$ using of *Matlab* (and *Mathematica*, too) for solving this problem can't give us satisfying results.

Implementation with *C++* - time

For the example we consider
(9 solutions, each with 16 components):

- average time of the algorithm *comparison with 0*: 1 μs;
- average time of the algorithm *bit comparison*: 0 μs.

$\Rightarrow$ this means that the time is in *ns*.

Comparison of The Times When We Have More Solutions

Number of Solutions	Time (μs) - C++		Time (μs) - Matlab	
	Bit Comparison	Comparison with 0	Bit Comparison	Comparison with 0
9	0	1	29	28
801	15	18	560	597
1601	31	39	1090	1113
8001	186	237	5377	5454

References

- Krassimir Danov and others.
  Effect of the Precipitation of Neutral-Soap, Acid-Soap, and Alkanoic Acid Crystallites on the Bulk pH and Surface Tension of Soap Solution

- Krassimir Danov and others.
  Coexistance of micelles and crystallites in solutions of potassium myristate: Soft matter vs. solid matter

- Krassimir Danov and others.
  Miccele-monomer equilibra in solutions of ionic surfactants and in ionic-nonionic mixtures: A generalized phase separation model

References

K. Birdi.
Surface and Colloid Chemistry: Principles and Applications, 2009.

Peter Atkins, Julio de Paula.
Physical Chemistry, 9th Edition , 2009.

Преслав Наков и Панайот Добриков.
Програмиране=++Алгоритми; Трето издание, 2005 г.

THANK YOU FOR YOUR ATTENTION!!!