Revista Mexicana de Física
ISSN: 0035-001X
rmf@ciencias.unam.mx
Sociedad Mexicana de Física A.C.
México

Lazzús, J.A.; Rivera, M.
Gas-solid phase equilibrium of biosubstances by two biological algorithms
Revista Mexicana de Física, vol. 59, núm. 6, noviembre-diciembre, 2013, pp. 577-583
Sociedad Mexicana de Física A.C.
Distrito Federal, México

RESEARCH Revista Mexicana de Física **59** (2013) 577–583 NOVEMBER-DECEMBER 2013

Gas-solid phase equilibrium of biosubstances by two biological algorithms

J.A. Lazzús and M. Rivera
Departamento de Física, Universidad de La Serena,
Casilla 554, La Serena, Chile.
e-mail: jlazzus@dfuls.cl

Received 8 May 2013; accepted 14 August 2013

Particle swarm optimization (PSO) and genetic algorithm (GA) are applied to the gas-solid phase equilibrium of biosubstances and to estimate their sublimation pressures (P^s). Four binary systems of supercritical carbon dioxide + biosubstances are considered in this study. The Peng–Robinson equation-of-state with the Wong–Sandler mixing rules, are used as a thermodynamic model to evaluate the fugacity coefficients in the classical solubility equation, and the van Laar model was incorporated to evaluate the excess Gibbs free energy included in the mixing rules. Then, the P^s is calculated from regression analysis of solubility data (y). P^s is usually small for most solid biosubstances and in many cases available experimental techniques cannot be used to obtain accurate values. Therefore, estimation methods must be used to obtain these data. PSO and GA are used for minimize the difference between calculated and experimental solubility. Comparing PSO with GA, it is shown that the results of PSO are better than that of GA, and provide a preferable method to estimate y and P^s of any biosubstances with high accuracy.

Keywords: Sublimation pressure; biosubstances; gas-solid equilibrium; equation of state; genetic algorithm; particle swarm optimization.

PACS: 51.30.+i; 64.75.Cd; 02.60.Pn

1. Introduction

In many cases of industrial fluid-solid separation process design, a thermodynamic key parameter may be the sublimation pressures P^s of pure components. The estimation of P^s may be a difficult problem in several chemical processes involving diverse substances, since nearly no experimental determinations are available under 0.1 Pa. A new trend in chemical applications is the use of supercritical solvents either in purifying operations on mixtures of complex biosubstances or in stripping on polluted stuff [1]. The optimization of this process can be performed only if the dependence of the solubility both on pressure and temperature can be accurately described and for this purpose, the P^s is considered as a key and relevant property [2].

Measurements of very low P^s of heavy components are very difficult to perform although their values are of most importance in the process evaluation [3]. P^s is relatively small as compared to vapor pressures of liquids at room temperature. In addition to that, experimental techniques cannot in many cases be used to accurately obtain P^s [2].

One of the common approaches used in the literature to correlate and predict phase equilibrium requires an equation of state that well relates the variables temperature, pressure and volume and appropriate mixing rules to express the dependence of the equation of state parameters on concentration [1] proposed to estimate P^s using the Peng–Robinson equation [4] and fusion properties.

Some researchers have estimated P^s from solubility data using an equation of state (EoS) and mixing rules to evaluate the fugacity [5,3]. For some substances, generalized correlations based on the molecular mass and the melting temperature have been proposed [6]. Also, to directly estimate the sublimation pressure, extrapolation of Antoine type equations have been used by Iwai et al. [7] and by Trabelsi *et al*. [8], while the Lee–Kesler equation have been employed by Nanping *et al*. [9]. Group contribution methods were applied by Coutsikos *et al*. [10] for predicting the vapor pressures of a variety of organic solids. Goodman *et al*. [11] obtained solid vapor enthalpy from functional groups and molecular parameters.

Phase equilibrium calculations of a solid dissolved in a compressed gas, at a pressure P and a temperature T can be performed using the fundamental equation of phase equilibria which leads to a simple equation that relates the solubility y, the pressure and the temperature. The equation also contains other properties such as P^s, the molar volume of the solid V^s and the fugacity coefficient of the solid component in the high pressure gas φ^s [2]. Of all these properties involved in the calculation of the solubility of the solid in the high pressure gas, P^s has received low attention in the literature, although it is directly related to the solubility. The molar volume does not have a strong influence on the calculations and the fugacity coefficient can be estimated from an appropriate equation of state and mixing rules [3].

In this study, four binary gas-solid phase systems of supercritical carbon dioxide + biosubstance were evaluated. The Peng–Robinson (PR) equation of state [4] was incorporated into the classical solubility equation. The Wong–Sandler (WS) mixing rules were used, and the van Laar model was included to evaluate the excess Gibbs free energy that appears in this mixing rule. Genetic algorithm [12], and particle swarm optimization [13,14] were programmed in C++, and used to minimize the difference between calculated and experimental solubility. Then, P^s of solid biosubstances were calculated from solubility data. This article considered the accuracy of two optimization algorithms in determining the sublimation pressure. In theory, the optimum sublimation pressures should be obtained by algorithm with the best performance in the minimization of the objective function, but this statement must be proved. Considering the valuable information which may be derived either directly or indirectly from sublimation data, it is rather surprising that there is so little quantitative information available in the literature on the sublimation process.

2. Gas-solid phase calculations

The experimental data for the supercritical carbon dioxide + biosubstances systems were correlated with the Peng–Robinson (PR) EoS [4], and the Wong–Sandler (WS) mixing rules [15].

The PR EoS was expressed as follows [4]:

$$P = \frac{RT}{V-b} + \frac{a}{V(V+b)+b(V-b)} \tag{1}$$

$$a = 0.457235 \frac{R^2 T_C^2}{P_C} \alpha(T_r) \tag{2}$$

$$b = 0.077796 \frac{RT_C}{P_C} \tag{3}$$

$$\alpha(T_r) = \left[1 + \kappa\left(1 - \sqrt{T_r}\right)\right]^2 \tag{4}$$

$$\kappa = 0.37646 + 1.54226\omega - 0.26992\omega^2 \tag{5}$$

where P, T and V are the pressure, temperature and molar volume respectively, $T_r = T/T_C$ is the reduced temperature. In this form, the PR EoS is completely predictive once the constants (critical temperature T_C, critical pressure P_C, and acentric factor ω) are given. Consequently, this equation is a two-parameter EoS (a and b) that depends upon the three constants (T_C, P_C, and ω).

And for mixtures:

$$P = \frac{RT}{V-b_m} + \frac{a_m}{V(V+b_m)+b_m(V-b_m)} \tag{6}$$

The WS mixing rules for the PR EoS that are used in this study can be summarized as follows [16]:

$$b_m = \frac{y_1^2\left(b-\frac{a}{RT}\right)_1 + 2y_1y_2\left(b-\frac{a}{RT}\right)_{12} + y_2^2\left(b-\frac{a}{RT}\right)_2}{1-\frac{y_1a_1}{b_1RT}-\frac{y_2a_2}{b_2RT}+\frac{A_\infty^E(y)}{\Omega RT}} \tag{7}$$

$$\left(b-\frac{a}{RT}\right)_{12} = \frac{1}{2}(b_1+b_2) - \frac{\sqrt{a_1a_2}}{RT}(1-k_{12}) \tag{8}$$

$$a_m = b_m\left(\frac{y_1a_1}{b_1RT} + \frac{y_2a_2}{b_2RT} + \frac{A_\infty^E(y)}{\Omega}\right) \tag{9}$$

In these equations a_m and b_m are the equation of state constants with k_{12} as adjustable parameter, $\Omega = 0.34657$ for the PR EoS, and $A_\infty^E(y)$ is calculated assuming that $A_\infty^E(y) \approx A_0^E(y) \approx G_0^E(y)$. For the excess Gibbs free energy $G_0^E(y)$, is calculated using an appropriate liquid phase model. In this study, $G_0^E(y)$ has been calculated using the van Laar model that has been shown to perform well in high pressure phase equilibrium calculations [3,17]. The van Laar model (VL) of $G_0^E(y)$ for a binary mixture is reduces to:

$$\frac{G_0^E}{RT} = \frac{\left(\frac{A_{12}}{RT}\right)y_1y_2}{y_1\left(\frac{A_{12}}{A_{21}}\right)+y_2} \tag{10}$$

The modeling of the solubility of a solid solute y_2 at equilibrium in a supercritical phase requires solving the following equilibrium condition [2],

$$y_2 = \frac{P_2^s\varphi^{sat}}{P\varphi_2^{SCF}} \exp\left[\frac{V_2^s(P-P_2^s)}{RT}\right] \tag{11}$$

where P_2^s is the sublimation pressure of the pure substance, V_2^s is the solid molar volume, and φ_2 is the fugacity coefficient of solid at the pressure P [2]. The fugacity coefficient is calculated from standard thermodynamic relations as [16]:

$$RT\ln\varphi_i = \int_V^\infty \left[\left(\frac{\partial P}{\partial n_i} - \frac{RT}{V}\right)_{T,V,n}\right] dV - RT\ln Z \tag{12}$$

The PR EoS [4] with the mixing rules proposed by Wong–Sandler [15] are used as the thermodynamic model to evaluate φ_2. The problem is then reduced to determining the parameters A_{12} and A_{21} in the van Laar model, the k_{12} parameter included in the combining rule for $(b - a/RT)_{12}$, and P_2^s that appears in the solubility equation, using available high pressure $T-P-y$ data for gas-solid systems as:

$$f = \frac{100}{N_D}\sum_{i=1}^{N_D}\left|\frac{y_2^{calc}-y_2^{exp}}{y_2^{exp}}\right|_i \tag{13}$$

where N_D is the number of points in the experimental data set and y_2 is the solid solute concentration in the gas phase, the superscript denotes the experimental (exp) data point and calculated ($calc$) values.

3. Particle swarm optimization (PSO)

In a PSO system, each particle is "flown" through the multidimensional search space, adjusting its position in search space according to its own experience and that of neighboring particles [3]. The particle therefore makes use of the best position encountered by itself and that of its neighbors to position itself toward an optimal solution [17]. Then the best position p_i is determined by [3,17]:

$$p_i(t+1) = \begin{cases} p_i(t) & \text{if } f(x_i(t+1)) \geq f(p_i(t)) \\ x_i(t+1) & \text{if } f(x_i(t+1)) < f(p_i(t)) \end{cases} \tag{14}$$

In each iteration, every particle calculates its velocity according to the following formula [3]:

$$\begin{aligned} v_i(t+1) &= wv_i(t) + c_1r_1\left(p_i(t) - x_i(t)\right) \\ &\quad + c_2r_2\left(p_g(t) - x_i(t)\right) \end{aligned} \tag{15}$$

where t is the current step number, w is the inertia weight, c_1 and c_2 are the acceleration constants, and r_1, r_2 are element from two random sequences in the range (0,1). The current position of the particle is determined by $x_i(t)$; p_i is the best one of the solutions this particle has reached, p_g is the best one of the solutions all the particles have reached [3,17].

The following weighting function w is used in Eq. (15):

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{t_{\max}} t \tag{16}$$

Generally, the value of each component in v can be clamped to the range $[-v_{\max}, v_{\max}]$ control excessive roaming of particles outside the search space [14]. After calculating the velocity, the new position of every particle is [3]:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{17}$$

The PSO algorithm performs repeated applications of the update equations above until a specified number of iteration has been exceeded, or until the velocity updates are close to zero [14].

The step-by-step to calculate the sublimation pressure using the particle swarm algorithm are described in details. To start the PSO the algorithm parameters must be defined, as: number of iteration $t = 100$, number of particles in the swarm $N_I = 50$, cognitive component $c_1 = 1.49$, social component $c_2 = 1.49$, maximum velocity $v_{\max} = 12$, minimum inertia weight $w_{\min} = 0.4$, and maximum inertia weight $w_{\max} = 0.41$. The search space of the PR-WS-VL model are defined, as `SS1` for k_{12} (0.02, 0.20), `SS2` for A_{12} $(1, 8)$, `SS3` for A_{21} $(1, 2)$, and `SS4` for $-\log P^s$ $(1, 5)$. 1) The initial population is created. This consists of $N_I = 50$ particles (individuals), associated to one position x_i. Let `XVect` and `VVect` denote a particle position and its corresponding velocity in a seach space, respectively. Each block represents the possible values of the parameters k_{12}, A_{12}, A_{21}, and P^s. 2) The vector containing the current position of particle `PVect`, is created. 3) The objective function is evaluated for each of the $N_I = 50$ sets of values $[k_{12}, A_{12}, A_{21}, P^s]$ and the values of the objective function are called the fitness. The evaluation of the objective function is done by calculating the solubility by the PR-WS-VL model for each value of the set of parameters. Calculated and experimental solubility values are introduced into the objective function resulting a vector of 50 elements, called `FunObj`. The elements are in this case the deviations between experimental and calculated values of the solubility, defined by Eq. (13). 4) The elements of the vector `FunObj` are arranged according to their decreasing fitness. Then, a new vector named `PBest` is formed. The new particle position is computed by adding the velocity vector to the current position Eq. (17). The velocity update equation is given by Eq. (16). 5) After calculating the velocity, the particle swarm algorithm performs repeated applications of the update equations above until a specified number of iteration has been exceeded $t_{\max} = 100$, or until the velocity updates are close to zero. 6) The matrix `VSol` is formed after evaluation process, and contains the individuals representing the optimum value of the parameters k_{12}, A_{12}, A_{21}, and P^s. The elements of the matrix are converted so that all numbers are in the search space for each of the variables (`SS1`, `SS2`, `SS3` and `SS4`). The set of parameters that gives the minimum value of the objective function represent the solution to the problem.

4. Genetic algorithm (GA)

Among the global stochastic optimization techniques, the evolutionary algorithms known as genetic algorithms have found many applications in several fields in science and engineering [18]. GA was first developed by Holland [12], and based on the mechanics of natural selection in biological systems. It uses a structure to utilize genetic information in finding new directions of search. The major genetic operators that reflect the nature of evolutionary process are reproduction, crossover and mutation [19].

GA maintains a population of individuals, whose characteristics are encoded in a fixed-length bit string, modeling the biological genotype [17]. It is at the discretion of the programmer the way in which these bits represent the phenotype (ontogeny). As a parallel to nature, genetic material is swapped between the individuals and mutated to produce offspring, with corresponding changes in their phenotypic performance. The crossover operator is an analogue of the recombination of genetic material as observed in reproduction. Crossover involves splitting the genomic bit-strings of two parents at a given number of locations and then splicing together complementary sections of each parent's bit-string to form the genotype of the new individual. Crossover occurs with a random probability. The mutation operator simulates natural mutation of DNA. This simply involves flipping bits in the string in a stochastic manner. Mutation should be fairly infrequent and should be applied following crossover [18].

The most significant differences between genetic algorithms and more traditional search and optimization methods, are: i) genetic algorithms search a population of points in parallel, not a single point: ii) genetic algorithms do not require derivative information or other auxiliary knowledge; only the objective function and the corresponding fitness levels influence the directions of search; iii) genetic algorithms use probabilistic transition rules, not deterministic ones; and iv) genetic algorithms work on an encoding environment of the parameter set rather than the parameter set itself [3].

The step-by-step to calculate the sublimation pressure using the genetic algorithm are described in details. To start the GA the algorithm parameters must be defined, as: number of iteration $it = 100$, number of individuals N_I=50, crossover operator $multipoint$, crossover probability $P_{cros} = 0.8$, mutation operator $binary$, and mutation probability P_{mut}=0.035. The search space of the PR-WS-VL model are defined, as `SS1` for k_{12} (0.02, 0.20), `SS2` for A_{12} (1,8), `SS3` for A_{21} (1,2), and `SS4` for $-\log P^s$ (1,5). 1)

The initial population is created. This consists of $N_I = 50$ individuals, represented by $N_C = 4$ chromosomes of $N_G = 20$ genes. Each block represents the possible values of the parameters k_{12}, A_{12}, A_{21}, and P^s. There are then 80 binary digits (0 or 1) randomly determined, so the total length is $N_{CG} = 80$. This forms the chromosome matrix `MatChrom` containing $[N_I \times N_{CG}]$ elements. 2) The elements of the matrix `MatChrom` are converted into real numbers in the established range (search space) for k_{12}, A_{12}, A_{21}, and P^s. This is done using a new Phenotype matrix `MatPhen`, containing $[N_I \times N_C]$. 3) The objective function is evaluated for each of the $N_I = 50$ sets of values $[k_{12}, A_{12}, A_{21}, P^s]$ and the values of the objective function are called the fitness. The evaluation of the objective function is done by calculating the solubility by the PR-WS-VL model for each value of the set of parameters. Calculated and experimental solubility values are introduced into the objective function resulting a vector of 50 elements, called `FunObj`. The elements are in this case the deviations between experimental and calculated values of the solubility, defined by Eq. (13). 4) The elements of the vector `FunObj` are arranged according to their decreasing fitness. The element whose value is the closest to cero is assigned an arbitrary value of 2.0. The element whose value is the highest in of the vector `FunObj` is assigned an arbitrary value of 0.0. The other $N_I - 2$ elements are linearly distributed between 0 and 2 according to their values (fitness). Thus, a new vector named `FunRank` is formed. 5) The vector `FunRank` and the `MatChrom` matrix determine the selection of the individuals for crossover. This consists of a random sampling in which the individual in the `MatChrom` matrix whose corresponding value in the `FunRank` vector are higher have more probability of being selected to be parents and reproduce offspring. It could happen that some individuals are selected more than once for reproduction. The new arrangement of the individuals, form a new matrix named `MatSel`. 6) Reproduction of the elements of the `MatSel` matrix is performed by multipoint Crossover. Crossover is done by interchanging some of the genes of one parent with some of the other parent. Crossover points are randomly defined and genes are interchanged. After this is done a new population of offsprings is created, population represented by the matrix `MatRecomb`. 7) Mutation is performed on the new population `MatRecomb`, according to the defined mutation probability ($P_{\text{mut}} = 0.035$). To do this, each gene is assigned a random number that represents the probability of being mutated. If the random number is less than P_{mut} the gene is selected for mutation. Mutation is done by changing the binary number (if it is 0 it is changed to 1 and if it is 1 it is changed to 0). The elements of the new matrix formed after mutations are randomly introduced into the chromosome matrix (`MatChrom`). 8) The new `MatChrom` represents a new generation of chromosomes, so steps 2 to 7 are repeated until all defined generations, have been created. 9) The matrix `MatPhen` is formed after evaluation process, and contains the individuals representing the optimum value of the parameters k_{12}, A_{12}, A_{21}, and P^s. The elements of the matrix `MatPhen` are converted so that all numbers are in the search space for each of the variables (`SS1`, `SS2`, `SS3` and `SS4`). The objective function is evaluated for each of the 50 sets $[k_{12}, A_{12}, A_{21}, P^s]$. The set of parameters that gives the minimum value of the objective function represent the solution to the problem.

5. Binary systems used

Four binary gas-solid phase systems of supercritical carbon dioxide + biosubstances containing: benzoic acid, caffeine, cholesterol, and β-carotene were considered in this study. These systems are of interest for the food, pharmaceutical and agrochemical industries, among others.

Table I shows the thermodynamic properties (V^s, T_C, P_C, and ω) of the substances used, and taken from Refs. 20

TABLE I. Thermodynamic properties of the substances involved in this study.

Substance	Formula	T_C(K)	P_C(MPa)	ω	V^s(cm^3/mol)	Ref.
Benzoic Acid	$C_7H_6O_2$	751.0	4.47	0.603	92.76	[20]
Caffeine	$C_8H_{10}N_4O_2$	855.6	4.15	0.555	145.68	[21]
Cholesterol	$C_{27}H_{46}O$	959.0	1.25	0.948	362.40	[20]
β-Carotene	$C_{40}H_{56}$	1177.0	1.24	1.040	536.50	[22]
Carbon dioxide	CO_2	304.2	7.38	0.224	—	[20]

TABLE II. Details on the phase equilibrium data of the four systems used in this study.

CO_2 +	ΔT(K)	N_D	ΔP(MPa)	Δy_2	Ref.
Benzoic Acid	308–328	33	10–36	(1.4–71.7)$\times 10^{-4}$	[23]
Caffeine	313–353	24	20–35	(2.8–11.3)$\times 10^{-4}$	[24]
Cholesterol	313–333	26	10–25	(0.02–1.5)$\times 10^{-4}$	[25]
β-Carotene	313–343	16	27.5–41.5	(0.9–25.4)$\times 10^{-8}$	[26]

to 22. The experimental phase equilibrium data taken from the literature: benzoic acid [23], caffeine [24], cholesterol [25], and β-carotene [26], are presented in Table II. As seen in the Table, the temperature and pressure ranges are narrow and go from 308 to 353 K and from 10 to 42 MPa, respectively.

For obtaining the optimum values of P^s using the solubility data in supercritical carbon dioxide, were calculated the binary interaction parameter k_{12}, A_{12} and A_{21} included in the PR-WS-VL model. These parameters are not directly related to P^s [3], but their values determine the acceptance of the model used as a good correlating tool for the solubility of the solid in the compressed gas phase [27,28].

6. Results and discussion

In this study, PSO and GA were programmed in C++, and used to minimize the difference between calculated and experimental solubility in Eq. (13). An exhaustive trial-and-error procedure has been applied for tuning the PSO and GA parameters.

The PR-WS-VL model and the biologically-deriver algorithms (PSO and GA) used to calculate P^s were evaluated by considering the deviations between experimental and calculated values of the solubility of the solid in the high pressure gas. The accurate modeling of the solubility is essential to obtain accurate values of P^s [3,17]. These deviations were expressed in relative form and absolute form, as follows:

$$|\%\Delta y_2| = \frac{100}{N_D}\sum_{i=1}^{N_D}\left|\frac{y_2^{calc}-y_2^{exp}}{y_2^{exp}}\right|_i \tag{18}$$

$$\%\Delta P^s = \frac{100}{N_D}\sum_{i=1}^{N_D}\left(\frac{P^s_{calc}-P^s_{exp}}{P^s_{exp}}\right)_i \tag{19}$$

$$|\%\Delta P^s| = \frac{100}{N_D}\sum_{i=1}^{N_D}\left|\frac{P^s_{calc}-P^s_{exp}}{P^s_{exp}}\right|_i \tag{20}$$

Table III shows the calculated binary interaction parameter A_{12} and A_{21} using PR-WS-VL model with the two biologically-deriver algorithms for the systems considered in this study. This Table shows the deviations of y and P^s for the four biosubstances using Eqs. (18-20). A comparison was made between of the results obtained with the PSO and GA algorithms and the results obtained with Levenberg–Marquart algorithm (LM) [29]. Note that, LM is commonly used in these problems. Table IV shows the mean values of the above variables of interest for these three algorithms. In this table, the best variables were calculated as an average of the best solution found by the three algorithms for all problems (four complex mixtures), and to evaluate the quality of the entire set of solutions that each algorithm provides. In general PSO performs better than GA and LM, with accuracy of 95 % and average deviation below than 5 %. The effect of number of individuals for PSO and GA on CPU time is show in Fig. 1. This figure illustrates the advantage of PSO in convergence time. The higher GA convergence time is due to the fact that GA needs decoding at each iteration. Figure 2 shows a comparison of the calculated binary interaction parameters for all mixtures, determined with PSO, GA and LM algorithms and based on the minimization of the objective function (Eq. (13)). The results found using PR-WS-VL model with biologically-deriver algorithms show that the gas solubility, and the sublimation pressures were correlated with low deviations between experimental and calculated values: the $|\%\Delta y|$ deviations were below than 15 %.

TABLE III. Interaction parameters, solubility, and sublimation pressure calculated with PSO and GA.

CO_2	T(K)	y_2	P^s_{exp}(Pa)	PSO estimation				GA estimation			
				A_{12}	A_{21}	$\|\%\Delta y\|$	$\|\%\Delta P^s\|$	A_{12}	A_{21}	$\|\%\Delta y\|$	$\|\%\Delta P^s\|$
Benzoic Acid	308	$(8.0–31.5)\times10^{-4}$	0.2117[20]	1.4545	0.6520	6.88	4.85	1.4956	0.8398	6.79	6.13
	318	$(2.6–48.7)\times10^{-4}$	0.7725[20]	1.2987	0.6555	8.07		1.2987	0.6554	8.31	
	328	$(1.4–71.7)\times10^{-4}$	2.4730[20]	1.4272	0.3299	4.83		1.5852	0.1814	5.30	
Caffeine	313	$(2.9–5.4)\times10^{-4}$	0.0496[21]	5.2444	9.6491	3.98	3.21	5.2884	8.4560	3.57	3.68
	333	$(3.0–7.2)\times10^{-4}$	0.6359[21]	6.1332	7.1497	3.30		6.1527	8.8879	3.37	
	353	$(2.8–11.3)\times10^{-4}$	4.9004[21]	6.8976	7.6983	9.50		6.8896	8.5069	14.53	
Cholesterol	313	$(0.1–1.0)\times10^{-4}$	2.63×10^{-4}[22]	6.9635	2.0779	2.02	2.24	6.7735	2.9879	3.22	3.32
	323	$(0.02–1.2)\times10^{-4}$	9.49×10^{-4}[22]	6.3920	1.9664	3.39		6.9940	1.9454	3.41	
	333	$(0.1–1.5)\times10^{-4}$	3.17×10^{-3}[22]	6.0294	0.4953	3.70		4.1016	0.7769	4.44	
β-Carotene	313	$(0.9–10.2)\times10^{-8}$	1.50×10^{-15}[23]	8.3909	7.4303	5.22	6.65	8.4845	8.4983	4.68	9.21
	333	$(4.3–27.2)\times10^{-8}$	9.81×10^{-14}[23]	10.330	8.2917	9.94		10.319	8.2339	10.95	
	343	$(19.5–25.4)\times10^{-8}$	6.52×10^{-13}[23]	7.1022	7.6303	9.05		7.1686	6.8413	11.08	
					Average		4.24		Average		5.59

TABLE IV. Mean values of the variables of interest for PSO, GA and LM.

Parameter	PSO	GA	LM
CPU time (s) –PC i3; RAM 4GB–	188	242	1079
Unique solutions in the final population (%)	90	72	—
Accuracy of solutions (%)	95.24	89.23	80.01
Minimun deviation (%)	0.72	1.55	5.99
Maximun deviation (%)	9.94	14.53	16.98
Average deviation (%)	4.24	5.59	10.33

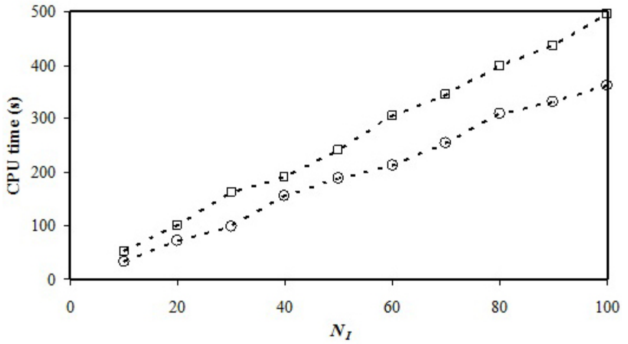

FIGURE 1. Effect of variation of the number of individuals on CPU time: (◯) PSO, and (□) GA.

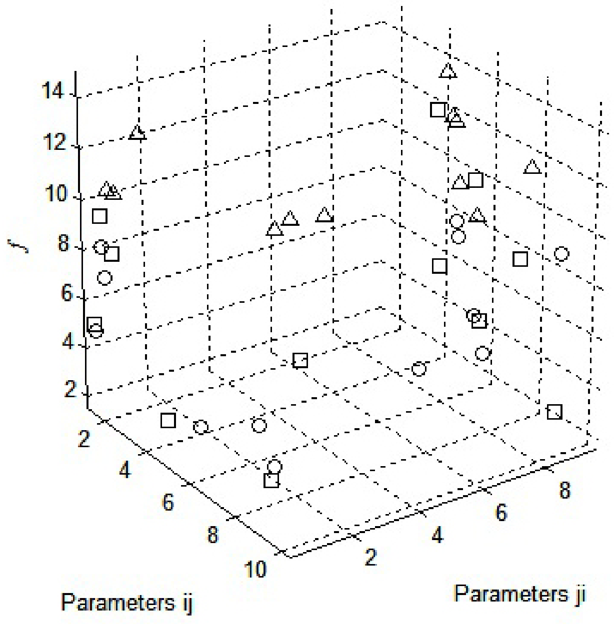

FIGURE 2. Deviations of the binary interaction parameters estimated by minimization with: (◯) PSO, (□) GA, and (△) LM.

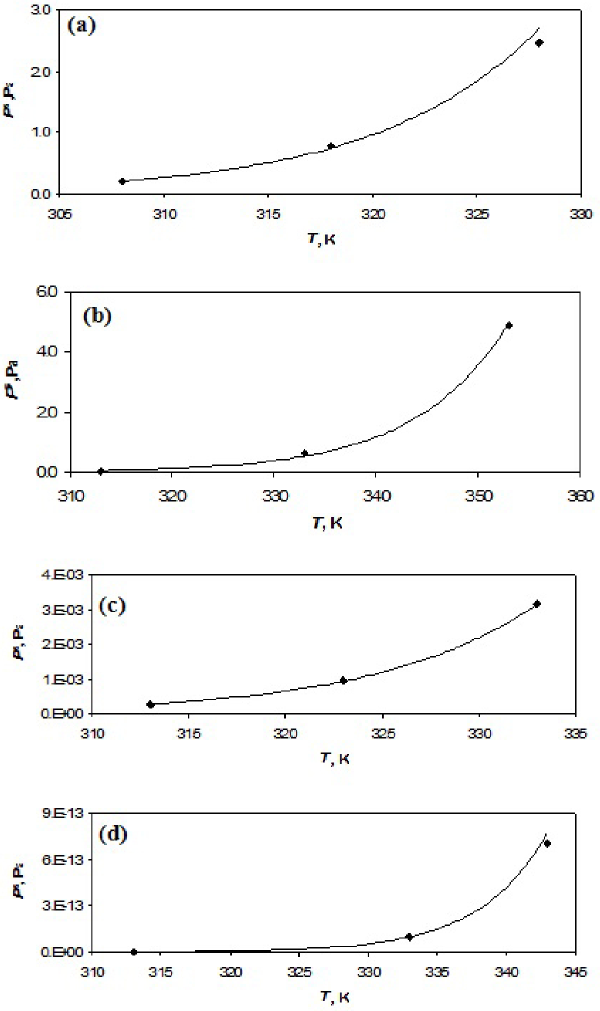

FIGURE 3. Sublimation pressures calculated with the PR-WS-VL model + PSO algorithm. In these figures: $solid\ lines$ are the experimental points and $filled\ diamonds$ are the calculated values. (a) benzoic acid, (b) caffeine, (c) cholesterol, and (d) β-carotene.

For the Ps calculations, the $\%\Delta P^s$ deviations were below than $\pm 15\%$, and the $|\%\Delta P^s|$ deviations were below than 10 %. Comparing the results obtained with PSO and GA, the best predictions of the P^s were calculated with the PSO algorithm (see, Table III). Figure 3 shows a visual general picture of the accuracy of the proposed PSO algorithm to calculate the P^s. The figures ratify the discussion presented above. These results indicate that the proposed method (PR-WS-VL + PSO) is reliable enough to estimate the P^s of any biosubstances using solubility data of the gas-solid systems containing supercritical fluids.

7. Conclusion

Based on the results and discussion presented in this study, the following main conclusions are obtained: i) the PR-WS-VL model is appropriate to modeling solid-gas phase equilibrium of supercritical carbon dioxide + biosubstance systems; ii) the PSO or GA are good tools for the calculation of the optimum value for the interaction parameters k_{12}, A_{12} and A_{21} using PR-WS-VL model; iii) in GA the following parameters are recommended for this type of calculations: number of iteration $it = 100$, number of individuals $N_I = 50$, crossover probability $P_{cros} = 0.8$, and mutation probability $P_{mut} = 0.035$; iv) in PSO the following parameters are recommended for this type of calculations: number of iteration $t = 100$, number of particles in the swarm $N_I = 50$, cognitive component $c_1 = 1.49$, social component $c_2 = 1.49$, maximum velocity $v_{\max} = 12$, minimum inertia weight $w_{\min} = 0.4$, and maximum inertia weight $w_{\max} = 0.41$; and v) the PSO showed to be a best algorithm to solve the optimization problem studied here.

Acknowledgments

The authors thank the Direction of Research of the University of La Serena (DIULS), and the Department of Physics of the University of La Serena (DFULS) for the special support that made possible the preparation of this paper.

---

1. E. Neau, S. Garnier and L.A. Avaullée, *Fluid Phase Equilib.* **164** (1999) 173.
2. J.A. Lazzús, *Thermochim. Acta* **489** (2009) 53.
3. J.A. Lazzús, *J. Eng. Thermophys.* **18** (2009) 306.
4. D.Y. Peng and D.B. Robinson, *Ind. Eng. Chem. Fundam.* **15** (1976) 59.
5. M. Zhong, B. Han, J. Ke, H. Yan and D.Y. Peng, *Fluid Phase Equilib.* **146** (1998) 93.
6. A. Vetere, *Fluid Phase Equilib.* **148** (1998) 83.
7. Y. Iwai, M. Yamamoto, Y. Hata, Y. Koga and Y. Arai, *J. Chem. Eng. Jpn.* **29** (1996) 728.
8. F. Trabelsi, K. Abaroudi and F. Recasens, *J. Supercrit. Fluids* **14** (1999) 151.
9. X. Nanping,W. Zhoohui, D. Junhang and S. Jun, *Chin. J. Chem. Eng.* **5** (1997) 29.
10. P. Coutsikos, E. Voutsas, K. Magoulas and D.P. Tassios, *Fluid Phase Equilib.* **207** (2003) 263.
11. B.T. Goodman, W.V. Wilding, J.L. Oscarson and R.L. Rowley, *Int. J. Thermophys.* **25** (2004) 337.
12. J. Holland, *Adaptation in Natural and Artificial Systems* (University of Michigan Press, USA, 1975).
13. R.C. Eberhart and J. Kennedy, *A new optimizer using particle swarm theory*. In: Proceedings of 6th International Symposium on Micro Machine and Human Science (IEEE Publishing, Nagoya, 1995) pp. 39-43.
14. J. Kennedy, R.C. Eberhart and Y. Shi, *Swarm Intelligence* (Academic Press, USA, 2001).
15. D.S. Wong and S.I. Sandler, *AIChE J.* **38** (1992) 671.
16. H. Orbey and S.I. Sandler, *Modeling Vapor-Liquid Equilibria. Cubic Equations of State and Their Mixing Rules* (Cambridge University Press, USA, 1998).
17. L. Palma-Chilla, J.A. Lazzús and A.A. Pérez Ponce, *J. Eng. Thermophys.* **20** (2011) 487.
18. L. Davis, *Handbook of Genetic Algorithms* (Van Nostrand Reinhold, New York, 1991).
19. K.W. Kim, Y. Yun, J. Yoon, M. Gen and G. Yamazaki, *Comput. Ind.* **56** (2005) 143.
20. T.E. Daubert, R.P. Danner, H.M. Sibul and C.C. Stebbins, *Physical and Thermodynamic Properties of Pure Chemicals. Data Compilation* (Taylor and Francis, London, 2000).
21. G. Xu, A.M. Scurto, M. Castier, J.F. Brennecke and M. Stadther, *Ind. Eng. Chem. Res.* **39** (2000) 1624.
22. P. Subra, S. Castellani, H. Ksibi and Y. Gabarros, *Fluid Phase Equilib.* **131** (1997) 269.
23. W.J. Schmitt and R.C. Reid, *J. Chem. Eng. Data* **31** (1986) 204.
24. M. Johannsen and G. Bruner, *Fluid Phase Equilib.* **95** (1994) 215.
25. S.L.J. Yun, K.K. Liong, G.S. Gurdial and N.R. Foster, *Ind. Eng. Chem. Res.* **30** (1991) 2476.
26. M.L. Cygnarowicz, R.J. Maxwell and W.D. Seider, *Fluid Phase Equilib.* **59** (1990) 57.
27. H. Singh, S.L. Yun, S.J. Macnaughton, D. Tomasko and N. Foster, *Ind. Eng. Chem. Res.* **32** (1993) 2841.
28. M. Skerget and Z. Knez, *J. Agric. Food Chem.* **45** (1997) 2066.
29. M. Reilly, *Computer Programs for Chemical Engineering Education* (Sterling Swift, Texas, 1972).