Swarm Intelligence Approach for Rational Global Approximation of Characteristic Curves for the Van der Waals Equation of State

Almudena Campuzano[1,2] , Andrés Iglesias[3,4(✉)] , and Akemi Gálvez[3,4]

[1] Faculty of Social and Behavioural Sciences, University of Amsterdam,
Amsterdam, The Netherlands
campuzanoalmudena@gmail.com
[2] School of Chemical, Biological and Environmental Engineering (CBEE),
College of Engineering, Oregon State University, Corvallis, OR, USA
[3] Department of Applied Mathematics and Computational Sciences,
University of Cantabria, Santander, Spain
{iglesias,galveza}@unican.es
[4] Department of Information Science, Faculty of Sciences, Toho University,
Funabashi-shi, Chiba-ken, Japan

Abstract. The Van der Waals (VdW) equation of state is a popular generalization of the law of ideal gases proposed time ago. In many situations, it is convenient to compute the characteristic curves of the VdW equation of state, called binodal and spinodal curves. Typically, they are constructed through data fitting from a collection of data points represented in the two-dimensional pressure-volume plane. However, the resulting models are still limited and can be further enhanced. In this paper, we propose to extend this polynomial approach by using a rational function as a fitting function. In particular, we consider a rational free-form Bézier curve, which provides a global approximation to the shape of the curve. This rational approach is more flexible than the polynomial one owing to some extra parameters, the weights. Unfortunately, data fitting becomes more difficult as these new parameters have also to be computed. In this paper we address this problem through a powerful nature-inspired swarm intelligence method for continuous optimization called the bat algorithm. Our experimental results show that the method can reconstruct the characteristic curves with very good accuracy.

Keywords: Equation of state · Characteristic curves · Van der Waals equation · Data fitting · Rational curves · Bat algorithm

1 Introduction

Equations of state (EoS) are key in several fields such as physics, thermodynamics, chemical engineering and many others. Roughly speaking, they are algebraic

expressions that describe the relation between physical variables such as temperature, T, pressure, P, and volume, V, for a blend or a component. Therefore, they can be used as predictors of the behavior of thermodynamic systems under different conditions [14].

Different EoS have been defined in the literature. The simplest and most widely known is the ideal gas law, given by:

$$P.V = n.R.T = \frac{m}{M}.R.T \tag{1}$$

where the mass, the molar mass and the number of moles are represented by the variables m, M and n, respectively. As usual, R represents the universal gas constant, whose value was used as R = 0.082 L.atm.mol^{-1}.K^{-1}. This equation is adequate for high temperatures and low pressures (i.e. about 1 atm). Nonetheless, incurs inaccuracies when applied to other conditions. Consequently, the equation has been modified throughout the years. A modification was proposed in 1873 by Johannes D. Van der Waals. This modification introduces two real, positive parameters, a and b, that account for the forces of interaction between molecules and molecular size, respectively [10]. This equation is referred to as the *Van der Waals (VdW) Equation of State* and is expressed as follows:

$$\left[P + \frac{a}{V_m^{\,2}} \right] (V_m - b) = R.T \tag{2}$$

where V_m is the molar volume. By setting affix the temperature T in (2), and plotting the variables of pressure P vs. volume V, an isotherm can be obtained (go to Sect. 2 for details). A phase diagram can be built by displaying and analysing isotherms for several values of T. This diagram will set the boundaries of the different regions of solid, liquid, and gas phases. These boundaries are defined by curves of non analytic behavior, and indicate the limit in which phase transitions take place. In particular, the gas-liquid transition can be explained by the construction of the characteristic curves: the *binodal curve*, beneath which two different phases can coexist, and the *spinodal* curve, that defines the unstable region in a system. From now on and for convenience reasons these curves will be referred to as *binodal* and *spinodal* respectively.

In general, it is not possible to analytically calculate these curves. They have to be computed by performing data-fitting of a collection of 2-D points previously obtained for distinct isotherms. The sequence of data points that define the binodal consists of the roots placed farthest to the left, the critical point (in which liquid and gas phases are indiscernible), and the roots placed farthest to the right. On the other hand, the spinodal is defined, from right to left, by the local maxima, the critical point, and the local minima of the collection of isotherms. Numerical procedures are used to determine both sequences of points. Then, the characteristic curves are obtained using polynomial data fitting (see [1,2,4,5,12] for details).

The organization of this paper is the following: The problem to be solved is presented in Sect. 2 as a continuous, nonlinear optimization problem. Section 3

offers an insight into the swarm intelligence approach used in this work: the bat algorithm. Section 4 describes in detail the methodology proposed, and the experimental results are reported in Sect. 5. Finally, the main conclusions are presented and some ideas for future work are explored.

2   Problem to Be Solved

2.1   Background

In this paper, we consider the VdW EoS expressed as (2). With some algebraic manipulation and rearranging, it can be obtained an expression for the relation between V and P, for any given T. Hence, for a set of temperatures, being $T_1, T_2, \ldots, T_M$, their respective isotherms can be determined, resulting in a set of curves in the P–V plot. Logically, each temperature will have its associated isotherm. Multiplying the expression by $V_m{}^2/P$, and rearranging, the result is a cubic polynomial:

$$V_m{}^3 - \left(b + \frac{RT}{P}\right) V_m{}^2 + \frac{a}{P} V_m - \frac{ab}{P} = 0 \tag{3}$$

which will have one or three real roots. It will be the case that only one real root exists for values of temperature, T, larger than the critical value of T_c, known as the critical temperature, and being characteristic of each substance. The second case, three real roots, happens for temperatures lower than T_c, when the isotherms oscillate up and down. The isotherm corresponding to $T = T_c$ is associated with a triple root, which defines the *critical point*. The scope of this work will be centered around the case of $T < T_c$; the three real roots linked to each correspondent isotherms will be named as R_1, R_2 and R_3. It is worthwhile to mention that the end roots, R_1 and R_3 are respectively associated to the liquid phase and vapour phase.

Given an scenario in which a temperature $T < T_c$ is raised until meeting the critical value, $T = T_c$, it occurs that the molar volume of the saturated liquid is increased, while in the case of the saturated vapor, the molar volume decreases. This saturated states represent the boundary between the single-phase region (for liquid or vapour respectively), and the coexisting-phases region (liquid/vapour). Mathematically, this can be translated as the two end roots, R_1 and R_3, moving towards each other as the temperature is raised, until they merge in the critical point. This means that at the critical point, liquid and vapour are indistinguishable. The critical values associated with the VdW EoS of a gas are only dependent on the previously-mentioned a and b parameters. This can be proved as follows:

$$V_c = 3.b, \qquad P_c = \frac{a}{27b^2}, \qquad T_c = \frac{8a}{27bR} \tag{4}$$

Working with dimensionless variables by considering the reduced temperature, pressure and volume:

$$(T_r, P_r, V_r) = \left(\frac{T}{T_c}, \frac{P}{P_c}, \frac{V}{V_c}\right) \tag{5}$$

Note that the molar volume, V_m, now is referred to as V for simplicity. Substituting and rearranging terms, Eq. (3) becomes:

$$V_r^3 - \frac{1}{3}\left(1 + \frac{8T_r}{P_r}\right)V_r^2 + \frac{3}{P_r}V_r - \frac{1}{P_r} = 0 \qquad (6)$$

The isotherms for $T < T_c$ exhibit a surprising behavior: if the volume is decreased, then the pressure increases, falls, and then increases again, describing a fluctuation; suggesting that the pressure of certain molar volumes can decrease as a consequence of compressing the fluid, which is associated with a negative isothermal compressibility, and therefore, identified as an unstable phase.

One way to fix this deficiency was proposed by James Clerk-Maxwell in [11], and is now referred to as *Maxwell's construction*, or *Equal area rule*. Basically, it proposes solving the situation by tracing a horizontal line through the fluctuating curve, so that it connects the dew point and the bubble point, in a way that the areas enclosed between the curve and the horizontal line would be equal. This horizontal line is called *tie line*.

2.2 Binodal and Spinodal Curves: Defining the Data Points

Let us examine how are the data-points of the binodal and spinodal obtained. In the case of the binodal, the first step is to define a set of increasing temperatures $T_1 < T_2 < \ldots T_M < T_c$. For every temperature value, there will be a pressure, P_j^*, that will split up the isotherm in two halves of equal dimensions. The value of P_j^* is calculated applying an optimization procedure that, through iteration, identifies the P_j^* that ensures the minimal difference between both areas. To that end, it begins with an initial guess of $\tilde{P}_j$ and iterates until it converges. Once that the right value of pressure has been determined, it is possible to compute the roots, R_k^j, for $(k = 1, 2, 3)$, as the intersect between the isotherm for T_j and the horizontal line $P = P_j^*$. This assemblage of vapour and liquid roots will conform the binodal. Hence, the defining points of the binodal curve, $\mathcal{B}$, can be listed as:

$$\mathcal{B} = \left\{ \{(R_1^j, P_j^*)\}_j, (1,1), \{(R_3^{M+1-j}, P_{M+1-j}^*)\} \right\}_{j=1,\ldots,M} \qquad (7)$$

On the other hand, the spinodal is conformed by the collection of points that define the local minima, $\mathbf{l}_j$, the critical point, and local maxima, $\mathbf{L}_l$. Note that vectors appear in bold.

The above-mentioned local optima points can be obtained through various techniques. One of them is solving the derivative $dP/dV = 0$ and examining the second derivative's sign, d^2P/dV^2, at the obtained solutions. If negative, the point corresponds to a maximum; otherwise, it is a minimum. Therefore, the assemblage of the points conforming the spinodal curve, $\mathcal{S}$, is defined by:

$$\mathcal{S} = \{\{\mathbf{l}_j\}_j, (1,1), \{\mathbf{L}_j\}_j\}_{j=1,\ldots,M} \qquad (8)$$

2.3 Characteristic Curves: Data Fitting

With the two sets of data points established, the characteristic curves can be reconstructed using standard numerical routines for data fitting. Taking into account that the obtained data points are influenced by some disturbances such as irregular sampling or noise, approximation is preferable, frequently using least-squares optimization. In such scenario, the function to be minimized is the error functional Ξ, which is the squared sum of residuals. The residual for the i-th data is given by the difference between observed data, μ_i, and the fitted data, $\hat{\mu}_i$:

$$\Xi = \sum_{i=1}^{\chi} (\mu_i - \hat{\mu}_i)^2 \tag{9}$$

Here χ is the total amount of data, and fitted data are procured by a certain fitting model function φ. It is worthwhile to mention that the minimization is carried out on the free variables of φ. Here φ is presumed to be a polynomial of a determined degree. As previously discussed, this choice can be extended by considering rational curves, as described in next section.

2.4 Data Fitting with Rational Bézier Curves

A *free-form rational Bézier curve* $\mathbf{\Phi}(\tau)$ *of degree η* is defined as [12]:

$$\mathbf{\Phi}(\tau) = \frac{\displaystyle\sum_{j=0}^{\eta} \omega_j \mathbf{\Lambda}_j \phi_j^{\eta}(\tau)}{\displaystyle\sum_{j=0}^{\eta} \omega_j \phi_j^{\eta}(\tau)} \tag{10}$$

where $\mathbf{\Lambda}_j$ are vector coefficients called the *poles*, ω_j are their scalar weights, $\phi_j^{\eta}(\tau)$ are the *Bernstein polynomials of index j and degree η*, given by:

$$\phi_j^{\eta}(\tau) = \binom{\eta}{j} \tau^j (1-\tau)^{\eta-j}$$

and τ is the *curve parameter*, defined on the finite interval $[0, 1]$. By agreement, $0! = 1$. As mentioned earlier, vectors will be denoted in bold.

Considered a set of data $\{\mathbf{\Delta}_i\}_{i=1,\ldots,\chi}$ in $\mathbb{R}^{\nu}$ (usually $\nu = 2$ or $\nu = 3$), the goal is to achieve the rational Bézier curve $\mathbf{\Phi}(\tau)$ through a discrete approximation of the data $\{\mathbf{\Delta}_i\}_i$. To that end, it is necessary to compute all parameters of the approximating curve $\mathbf{\Phi}(\tau)$, (i.e. weights ω_j, poles $\mathbf{\Lambda}_j$, and parameters τ_i associated with data points $\mathbf{\Delta}_i$, for $i = 1, \ldots, \chi$, $j = 0, \ldots, \eta$), by minimizing the least-squares error, Υ, defined as the sum of squares of the residuals:

$$\varUpsilon = \underset{\substack{\{\tau_i\}_i \\ \{\mathbf{\Lambda}_j\}_j \\ \{\omega_j\}_j}}{\text{minimize}} \left[\sum_{i=1}^{\chi} \left(\mathbf{\Delta}_i - \frac{\displaystyle\sum_{j=0}^{\eta} \omega_j \mathbf{\Lambda}_j \phi_j^{\eta}(\tau_i)}{\displaystyle\sum_{j=0}^{\eta} \omega_j \phi_j^{\eta}(\tau_i)} \right)^2 \right]. \tag{11}$$

Now, taking:

$$\varphi_j^{\eta}(\tau) = \frac{\omega_j \phi_j^{\eta}(\tau)}{\displaystyle\sum_{k=0}^{\eta} \omega_k \phi_k^{\eta}(\tau)} \tag{12}$$

Eq. (11) becomes:

$$\varUpsilon = \underset{\substack{\{\tau_i\}_i \\ \{\mathbf{\Lambda}_j\}_j \\ \{\omega_j\}_j}}{\text{minimize}} \left[\sum_{i=1}^{\chi} \left(\mathbf{\Delta}_i - \sum_{j=0}^{\eta} \mathbf{\Lambda}_j \varphi_j^{\eta}(\tau) \right)^2 \right], \tag{13}$$

which can be rewritten in matrix form as: $\mathbf{\Omega}.\mathbf{\Lambda} = \mathbf{\Xi}$, where: $\mathbf{\Omega} = [\Omega_{i,j}] = \left[\left(\sum_{k=1}^{\chi} \varphi_i^{\eta}(\tau_k) \varphi_j^{\eta}(\tau_k) \right) \right]_{i,j}$, $\mathbf{\Xi} = [\Xi_j] = \left[\left(\sum_{k=1}^{\chi} \mathbf{\Delta}_k \varphi_j^{\eta}(\tau_k) \right) \right]_j$, $\mathbf{\Lambda} = (\mathbf{\Lambda}_0, \ldots, \mathbf{\Lambda}_\eta)^T$, for $i, j = 0, \ldots, \eta$, and $(.)^T$ means transposition.

Generally, $\chi >> \eta$, meaning that $\mathbf{\Omega}.\mathbf{\Lambda} = \mathbf{\Xi}$ is an overdetermined system of equations. If τ_i had assigned values, the problem could be solved by standard optimization procedures with coefficients $\{\mathbf{\Lambda}_i\}_{i=0,\ldots,\eta}$ as unknowns. However, since τ_i are treated as unknowns, the complexity of the problem escalates. In fact, as the polynomial blending functions $\phi_j^{\eta}(\tau)$ and the rational blending functions $\varphi_j^{\eta}(\tau)$, are nonlinear in τ, the least-squares minimization of the residuals turns to be a continuous, nonlinear optimization problem. It can also involve a large number of unknowns, since in reality the problem can present an extremely large amount of data points. Since there may not be only one unique set of parameters leading to the solution, the problem is also multimodal. On the whole, the complicated interplay among all the unknowns (data parameters, poles, and weights) leads to a highly complex overdetermined, continuous, multivariate, multimodal, nonlinear optimization problem. The aim of this work is to solve this problem. Instead of assuming certain values for some free parameters, they are all included in our computations. This problem cannot be solved by applying classical mathematical optimization techniques [4]. In this work we propose the application of the bat algorithm, a high-power evolutionary computational method, already successfully applied to other data-fitting optimization problems in previous works [7–9]. In the next section this algorithm is further discussed.

3 The Bat Algorithm

The *bat algorithm* is a computational intelligence algorithm devised for continuous optimization problems [19,20]. It is inspired by some particular features

of the social and motion behavior of small bats (microbats). These microbats use a particular kind of sonar called *echolocation* for different purposes, such as prey detection, obstacle avoidance, or roosting crevices detection, among others. Introduced in 2010, the bat algorithm has found remarkable applications for several problems [9,15–17]. See also [21] for a detailed review of the bat algorithm.

The bat algorithm is a population-based method in which the individuals (bats) are randomly initialized and distributed over the search space and then, they perform extensive exploration searching for the best location, a variable related to the quality of the solution. When a bat i is moving, its dynamics at iteration g is determined by its frequency f_i^g, location $\mathbf{x}_i^g$, and velocity $\mathbf{v}_i^g$. These variables are governed by the following evolution equations:

$$f_i^g = f_{min}^g + \beta(f_{max}^g - f_{min}^g) \tag{14}$$
$$\mathbf{v}_i^g = \mathbf{v}_i^{g-1} + [\mathbf{x}_i^{g-1} - \mathbf{x}^*]\, f_i^g \tag{15}$$
$$\mathbf{x}_i^g = \mathbf{x}_i^{g-1} + \mathbf{v}_i^g \tag{16}$$

where β is a uniform random variable on $[0, 1]$, and $\mathbf{x}^*$ is used to represent the current global best location (solution), obtained by evaluating the fitness function at all bats and then ranking the corresponding fitness values. The method then performs a local search in the neighborhood of the current best solution through a random walk of the form:

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \epsilon \mathcal{A}^g$$

with ϵ being a uniform random number on $[-1, 1]$ and where $\mathcal{A}^g = <\mathcal{A}_i^g>$, represents the average loudness of all the bats of the population at generation g. Any new solution that is better than the previous best solution is accepted with a certain probability that depends on the value of the loudness. In case of acceptance, the pulse rate is increased according to the law:

$$r_i^{g+1} = r_i^0[1 - exp(-\gamma g)]$$

where γ is a parameter of the method.

Simultaneously, the loudness is decreased, following an evolution rule:

$$\mathcal{A}_i^{g+1} = \alpha \mathcal{A}_i^g$$

with α being another parameter of the method. This procedure is repeated iteratively for a maximum number of iterations, given by a parameter $\mathcal{G}_{max}$.

It is generally assumed that each bat has different values for the loudness and the pulse emission rate. This is achieved by considering the initial values for the loudness randomly as $\mathcal{A}_i^0 \in (0, 2)$. The emission rate takes an initial random value r_i^0 in the interval $[0, 1]$. Both parameters are updated only when the new solutions are better than the current ones, which is interpreted as a sign that the bats are advancing towards the optimal global solution.

4   The Method

4.1   Overview of the Method

As explained above, the Van der Waals Equation of State in (2) introduces two parameters, a and b, characteristic of each chemical element. These two parameters, together with a set of temperatures $T_1 < T_2 < \ldots T_M$ below the critical temperature of the substance, T_c, are the starting input of the problem. Our method is comprised of the subsequent steps:

1. Compute V_c, P_c, T_c, the critical values, using (4).
2. Compute the reduced variables V_c, P_c, T_c with (5).
3. Compute isotherms at temperatures T_j from (2).
4. For every isotherm of T_j:
    4a. Contemplate a first guess $\tilde{P}_j$ and obtain the value of P_j^* through optimization, applying Maxwell's construction.
    4b. With P_j^*, compute the roots of (6), as recounted in Sect. 2.2.
    4c. Obtain the local optima of (6), as described in Sect. 2.2.
    The result of (4a.) and (4b.) will be the sets of data points, $\mathcal{B}$ and $\mathcal{S}$, for the binodal and the spinodal curves, found respectively in (7) and (8).
5. Apply rational Bézier curves for data fitting on $\mathcal{B}$ and $\mathcal{S}$ as indicated:
    5a. Obtain data parameterization for $\mathcal{B}$ and $\mathcal{S}$ and weight computation using the bat algorithm (further discussed in Sect. 4.2).
    5b. Compute the poles of the curve applying least-squares optimization. Resolve the equations system applying classical numerical procedures, such as singular value decomposition (SVD), standard LU decomposition, and a modification of the LU decomposition for non-squared sparse problems (see [13] for details).

The most important and crucial part of the method, as well as the key component of this paper is the step (5a), which will be discussed in the next section.

4.2   Bat Algorithm for Data Fitting

This section describes how the bat algorithm, presented in Sect. 3, is used for data parameterization and weight computation with rational Bézier curves. To this purpose, we need to consider:

1. Bat encoding. In our problem, the free variables are going to be represented as follows. Bats, being denoted by $\mathcal{B}_k$, are vectors of real numbers of length $M + \eta + 1$, corresponding to a parameterization of data points and the weights, as follows:

$$\mathcal{B}_k = (\rho_1^k, \rho_2^k, \ldots, \rho_M^k, \omega_0^k, \omega_1^k, \ldots, \omega_\eta^k) \qquad (17)$$

All bats $\{\mathcal{B}_k\}_k$ are initialized with uniformly distributed random numbers on the interval $[0, 1]$ for the ρ_j^k and with real positive values on the interval $(0, 20]$ for the ω_j^k. The $\{\rho_i^k\}_i$ are arranged in ascending order to reproduce the orderly form of data parameterizaton.

Table 1. Parameters of the bat algorithm and the values used in this work.

Notation	Explanation	Range	Selected value
$\mathcal{P}$	Size of the population	50–300	100
$\mathcal{G}_{max}$	Max. no. of iterations	200–3,000	1,000
$\mathcal{A}^0$	Initial loudness	$(0,2)$	0.5
$\mathcal{A}_{min}$	Minimum loudness	$[0,1]$	0
r^0	Initial pulse rate	$[0,1]$	0.2
f_{max}	Max. frequency	$[0,10]$	1.5
α	Multiplicative factor	$(0,1)$	0.3
γ	Exponential factor	$[0,1]$	0.2

2. Fitness function. It dovetails with the estimation of the least-squares function (11). However, as this function ignores the total number of data points, the RMSE (root-mean squared error) is also computed:

$$RMSE = \sqrt{\frac{\Upsilon}{\chi}} \tag{18}$$

3. Curve parameters. There is solely one parameter, which is the degree of the curve, η. This value will influence the amount of weighs and poles. In this work we empirically determined the optimal value, by computing and comparing the RMSE for different values of η, from 2 to 7.

4. Bat algorithm parameters. The algorithm has some key parameters that need to be tuned. This is of paramount importance for the proper functioning of the method. The task entails a challenge, because these parameters depend heavily on the problem. In this work the authors chose the best value by comparison from a vast set of empirical results, obtained after performing a large amount of simulations. The adjusted parameters are displayed in rows in Table 1, with their notation, explanation, and range arranged in columns, along with the final selected value. The parameters that are most decisive are the population size, $\mathcal{P}$, and the maximum number of iterations, $\mathcal{G}_{max}$. The size of population is set to $\mathcal{P} = 100$ in all shown cases. More extensive populations were also tested, up to 300, without any significant effect. As for the number of iterations, the bat algorithm results particularly beneficial since a number of $\mathcal{G}_{max} = 1000$ is sufficient to reach convergence, as opposed to other algorithms that typically requires a much larger number.

With the above-mentioned parameters selected, the bat algorithm is run for the fixed number of iterations. Last, the simulation with the top fitness value for (18) is chosen as the problem's best solution.

5   Experiments and Results

5.1   Application to a Real Case: Argon

We have applied our method to the Van der Waals (VdW) Equation of State for the case of argon, Ar. This noble gas is the third most abundant of its kind in the atmosphere, and has countless applications in industrial processes, research, medicine or lighting [18]. Its VdW parameters are $a = 1.355 \ atm.L^2.mol^{-2}$ and $b = 0.03201 \ L.mol^{-1}$. The value of the critical temperature is $T_c = 150.86$ K, with a margin of 0.1 K as reported by [3,6].

First to third steps from our workflow were performed for the following set of temperatures: $\{130, 133, 135, 137, 140, 142, 145, 147, 148, 149, T_c\}$ K and $\{128, 130, 133, 135, 137, 140, 142, 145, 147, 148, 149, 150.2, T_c\}$ K, respectively for the binodal and spinodal. Subsequently, the lists of data points for the characteristic curves, $\mathcal{B}$ and $\mathcal{S}$, were obtained following the fourth step. In the step *4a*, the Vandermonde matrix is used for carrying out the standard polynomial linear fitting of the optimization process. Data parameterization and weight computation are completed following the procedure in Sect. 4.2. The result is a linear system that can be solved using SVD. By doing so, pole computation is accomplished.

5.2   Computational Results

To account for stochastic effects and prevent premature convergence, 30 individual simulations were run for every value of η. The worst 10 executions were dropped to avoid the specious effects of instability. Computational results are reported in Tables 2 and 3 for binodal and spinodal curves respectively, with the degree ranging from $\eta = 2$ to $\eta = 7$ (in rows). We remark that, although our previous experiments for polynomial curves included values up to $\eta = 9$, the values $\eta > 7$ are actually unnecessary because of the extra degrees of freedom given by the weights. They also introduce large numerical errors, so values for η larger than 7 are finally discarded in our discussion. We have also compared our current results for the rational curves with the previous ones with strictly polynomial curves. The comparative results are displayed in Tables 2 and 3. Each table presents, in columns, the curve degree, the best RMSE (for the 30 executions), and the RMSE mean (for the 20 best executions) for the polynomial Bézier curves (columns 2 and 3) and for the rational Bézier curves (columns 4 and 5).

It can be seen from the good values of the fitting errors that the method performs pretty good. The RMSE, best and mean, achieve values of order as low as 10^{-4} for all degrees, except $\eta = 2$, meaning that the fail to be replicated with a quadratic curve, owing to the fact of not being parabolas. The best fitting rational curves are obtained for $\eta = 4$ for the binodal curve (although $\eta = 3$ performs almost equivalently) and for $\eta = 3$ for the spinodal curve (although the errors for $\eta = 4$ are also very similar). The RMSE degrees from $\eta = 4$ to $\eta = 9$, tend to be of the same order. This fact indicates that functions of higher degree are associated with more degrees of freedom (DOFs) and consequently,

Table 2. Computational results for the binodal curve.

Degree	Polynomial approach		Rational approach	
	RMSE (best)	RMSE (mean)	RMSE (best)	RMSE (mean)
$\eta = 2$	3.5349E−2	5.6509E−2	3.3873E−2	5.4522E−2
$\eta = 3$	7.7394E−5	9.8801E−5	6.5902E−5	7.7883E−5
$\eta = 4$	8.5572E−5	1.0833E−4	5.9717E−5	7.4588E−5
$\eta = 5$	1.0365E−4	1.1776E−4	8.4615E−5	1.0513E−4
$\eta = 6$	1.1927E−4	1.3328E−4	9.0631E−5	1.0264E−4
$\eta = 7$	1.1624E−4	1.2684E−4	9.0724E−5	1.0016E−4

Table 3. Computational results for the spinodal curve.

Degree	Polynomial approach		Rational approach	
	RMSE (best)	RMSE (mean)	RMSE (best)	RMSE (mean)
$\eta = 2$	4.9074E−2	7.1226E−2	4.3812E−2	6.2144E−2
$\eta = 3$	9.8649E−5	1.0356E−4	6.3187E−5	8.4551E−5
$\eta = 4$	1.1267E−4	1.2842E−4	7.3063E−5	8.5114E−5
$\eta = 5$	1.2060E−4	1.2953E−4	9.9672E−5	1.0724E−4
$\eta = 6$	1.4971E−4	1.8611E−4	1.1163E−4	1.3974E−4
$\eta = 7$	1.4750E−4	1.6975E−4	1.1708E−4	1.4828E−4

they achieve better fitting. Naturally, this occurs at the price of a higher model complexity, so in the case of numerical errors of a similar order, the values providing the simplest model are more desirable and should be selected with higher priority. In fact, an additional problem is that these extra degrees of freedom may cause over-fitting. Actually, this holds true for the spinodal and binodal respectively, for models of degree $\eta \geq 6$ and $\eta \geq 7$. Hence, the curves that must be considered predictive for other temperature values, are only those of low degree.

Another important observation is the excellent CPU times of around only 2∼4 min. On the other hand, simulations in alternative swarm intelligence methods can take as long as tens of minutes for a single execution. This advantage is owed to the quick convergence of this method. Such competitive computational times are a good indicator of the applicability of our method. We remark, however, that the CPU times for the rational case are still slightly larger (but not dramatically) than for the polynomial case, which is consistent with the fact that some extra free parameters have to be computed, thus requiring extra computation time in our simulations.

Regarding the implementation, the equipment used for all the computations was a 3.4 GHz. Intel Core i7 processor with 8 GB. of RAM. The authors implemented all the source code in *MATLAB*, version 2018b.

6   Conclusions and Future Work

In this manuscript, a new method to construct the characteristic curves of the Van der Waals equation of state through data fitting is presented. The method relies on the use of rational Bézier curves. Considering the parameters a and b of a chemical system as the input for our method, two sets of data points for the binodal and spinodal curves are firstly obtained; then, they are used to perform data parameterization and weight computation by means of the bat algorithm; finally, we apply least-squares optimization with singular value decomposition to compute the poles of the curves. The method is applied to a chemical element, argon. The method performs very well, and reconstructs the curves with high accuracy. Furthermore, it is reasonably fast (although slower than the polynomial case), with CPU times in the range of 2–4 min for each execution.

About the plans for future work, we wish to further improve the accuracy of our method. We also want to reduce the computational time of the method. We are also planning to apply this approach to other chemical components and mixtures, as well as extending this approach to other popular equations of state of interest in the field.

References

1. Campuzano, A., Iglesias, A., Gálvez, A.: Applying firefly algorithm to data fitting for the Van der Waals equation of state with Bézier curves. In: Proceedings of the International Conference on Cyberworlds (CW 2019), pp. 211–214. IEEE Computer Society Press, Los Alamitos (2019)
2. Campuzano, A., Iglesias, A., Gálvez, A.: Free-form parametric fitting of van der Waals binodal and spinodal curves with bat algorithm. In: Proceedings of the International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2019), pp. 1–8. IEEE Computer Society Press, Los Alamitos (2019)
3. Angus, S., et al.: International Thermodynamic Tables of the Fluid State - 1 Argon. Butterworths, London (1972)
4. Dierckx, P.: Curve and Surface Fitting with Splines. Oxford University Press, Oxford (1993)
5. Farin, G.: Curves and Surfaces for CAGD, 5th edn. Morgan Kaufmann, San Francisco (2002)
6. Gosman, A.L., McCarty, R.D., Hust, J.G.: Thermodynamic properties of Argon from the triple point to 300 K at pressures to 1000 atmospheres. Data Ser. Nat. Bur. Stand. **27**, 1–146 (1969)

7. Iglesias, A., Gálvez, A., Collantes, M.: Bat algorithm for curve parameterization in data fitting with polynomial Bézier curves. In: Proceedings of the Cyberworlds (CW 2015), pp. 107–114. IEEE Computer Society Press, Los Alamitos (2015)

8. Iglesias, A., Gálvez, A., Collantes, M.: Global-support rational curve method for data approximation with bat algorithm. In: Chbeir, R., Manolopoulos, Y., Maglogiannis, I., Alhajj, R. (eds.) AIAI 2015. IAICT, vol. 458, pp. 191–205. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23868-5_14

9. Iglesias, A., Gálvez, A., Collantes, M.: Multilayer embedded bat algorithm for B-spline curve reconstruction. Integr. Comput.-Aided Eng. **24**(4), 385–399 (2017)

10. Johnson, D.C.: Advances in Thermodynamics of the Van der Waals Fluid. Morgan & Claypool Publishers, Ames (2014)

11. Maxwell, J.C.: On the dynamical evidence of the molecular constitution of bodies. Nature **11**, 357–359 (1875)

12. Piegl, L., Tiller, W.: The NURBS Book, pp. 146–165. Springer, Heidelberg (1997). https://doi.org/10.1007/978-3-642-59223-2

13. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes, 2nd edn. Cambridge University Press, Cambridge (1992)

14. Smith, J.M., Van Ness, H.C., Abbott, M.M.: Introduction to Chemical Engineering Thermodynamics. McGraw-Hill, Boston (2005)

15. Suárez, P., Iglesias, A.: Bat algorithm for coordinated exploration in swarm robotics. In: Del Ser, J. (ed.) ICHSA 2017. AISC, vol. 514, pp. 134–144. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-3728-3_14

16. Suárez, P., Gálvez, A., Iglesias, A.: Autonomous coordinated navigation of virtual swarm bots in dynamic indoor environments by bat algorithm. In: Tan, Y., Takagi, H., Shi, Y., Niu, B. (eds.) ICSI 2017. LNCS, vol. 10386, pp. 176–184. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61833-3_19

17. Suárez, P., Iglesias, A., Gálvez, A.: Make robots be bats: specializing robotic swarms to the bat algorithm. Swarm Evol. Comput. **44**, 113–129 (2019)

18. Weast, R.C.: Handbook of Chemistry and Physics, 53rd edn. Chemical Rubber Publishing, Boca Raton (1972)

19. Yang, X.S.: A new metaheuristic bat-inspired algorithm. In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds.) NICSO 2010. SCI, vol. 284, pp. 65–74. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12538-6_6

20. Yang, X.S., Gandomi, A.H.: Bat algorithm: a novel approach for global engineering optimization. Eng. Comput. **29**(5), 464–483 (2012)

21. Yang, X.S.: Bat algorithm: literature review and applications. Int. J. Bio-Inspired Comput. **5**(3), 141–149 (2013)