A Comprehensive Analysis of Quantum Clustering : Finding All the Potential Minima

Aude Maignan, Tony Scott

A comprehensive analysis of Quantum clustering: Finding All the potential minima

Aude Maignan[1] and Tony Scott[2]

[1]Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

aude.maignan@univ-grenoble-alpes.fr

[2] Institut für Physikalische Chemie, RWTH-Aachen University, 52056 Aachen, Germany

scotusts@yahoo.com

ABSTRACT

Quantum clustering (QC), is a data clustering algorithm based on quantum mechanics which is accomplished by substituting each point in a given dataset with a Gaussian. The width of the Gaussian is a σ value, a hyper-parameter which can be manually defined and manipulated to suit the application. Numerical methods are used to find all the minima of the quantum potential as they correspond to cluster centers. Herein, we investigate the mathematical task of expressing and finding *all* the roots of the exponential polynomial corresponding to the minima of a two-dimensional quantum potential. This is an outstanding task because normally such expressions are impossible to solve analytically. However, we prove that if the points are all included in a square region of size σ, there is only one minimum. This bound is not only useful in the number of solutions to look for, by numerical means, it allows to to propose a new numerical approach "per block". This technique decreases the number of particles by approximating some groups of particles to weighted particles. These findings are not only useful to the quantum clustering problem but also for the exponential polynomials encountered in quantum chemistry, Solid-state Physics and other applications.

KEYWORDS

Data clustering, Quantum clustering, energy function, exponential polynomial, optimization

1. INTRODUCTION

The primary motivation for this work stems from an important component of the area of information retrieval of the IT industry, namely *data clustering*. For any data of a scientific nature such as Particle Physics, pharmaceutical data, or data related to the internet, security or wireless communications, there is a growing need for data analysis and predictive analytics. Researchers regularly encounter limitations due to large datasets in complex simulations, in particular, biological and environmental research. One of the biggest problems of data analysis is data with no known *a priori* structure, the case of "unsupervised data" in the jargon of machine learning. This is especially germane to object or name disambiguation also called the "John Smith" problem [1]. Therefore *data clustering*, which seeks to find internal classes or structures within the data, is one of most difficult yet needed implementations.

It has been shown that the quantum clustering method (QC) [2] [3] can naturally cluster data originating from a number of sources whether they be: scientific (natural), engineering and even text. In particular, it is more stable and is often more accurate than the standard data clustering method known as K-means [3]. This method requires isolating the minima of a quantum potential and is equivalent to finding the roots of

its gradients i.e. an expression made of exponential polynomials. Finding all the clusters within the data means finding *all* the potential minima. The quantum clustering method can be viewed as "dual" or inverse operation of the machine learning process known as a nonlinear support vector machines when using Gaussian functions are used as its kernel function; this machine learning approach being the very inspiration of the quantum clustering method [4].

This is not the only problem in quantum mechanics requiring such solutions. The nodal lines of any given wave function characterize it with respect to internal symmetries and level of excitation. In general, if one arranges the eigenstates in the order of increasing energies, e.g. ϵ_1 ,ϵ_2 , ϵ_3, …the eigenfunctions likewise fall in the order of increasing number of nodes; the n^{th} eigenfunction has $n-1$ nodes, between each of which the following eigenfunctions have at least one node [5]. In diffusion Monte-Carlo calculations for Molecules, a precise determination of the nodal structure of wave function yields greater accuracy for the energy eigenvalues [6] [7] [8]. Furthermore, solutions in terms of Gaussian functions involve the most developed mathematical "technology" of quantum chemistry (e.g. The Gaussian program [9]). This is not surprising for the following reasons:

1. In principle, we can get *all* the roots of polynomial systems. However, quantum mechanical systems need exponentials in order to ensure a square-integrable wave function over all space. About an atom, the angular components over a range $(0,2\pi)$ can be modeled in terms of polynomials of trigonometric quantities such as e.g. Legendre polynomials. However, the radial part extends over all space requiring exponential apodization.
2. Thanks to properties such as the Gaussian product theorem, Gaussian functions allow for exact analytical solutions of the molecular integrals of quantum chemistry [10] [11] [12].
3. In general, for small atoms and molecules, the nodal lines can be modeled as nodes of polynomial exponentials [13] [14] [15].

More recently, in the area of low temperature Physics (including superconductors), clustering within machine learning has been used in finding phases and separating the data into particular topological sectors [16] [17] [18]. High accuracy of the clustering is crucial in order to precisely identify transition points in terms of e.g. temperature or pressure.

To reiterate, any insight concerning the isolation of *all* the roots or nodal lines of polynomial exponentials is useful for quantum clustering and computational quantum chemistry and condensed matter Physics and data analysis. This has applications in all cases for any given function covering all space in principle but whose extrema and/or roots are in a finite local region of space.

1.1 Statement of the Problem

Consider a set of particles $(X_i)_{i=1..N}$, the quantum clustering is a process that detects the clusters of the distributed set $(X_i)_{i=1..N}$ by finding the cluster centers. Those centers are the minima of the potential energy function defined by [2] [3]:

$$\frac{1}{2\sigma^2}\frac{1}{\sum_{i=1}^{N} e^{-\frac{(X-X_i)^2}{2\sigma^2}}}\sum_{i=1}^{N}(X-X_i)^2 e^{-\frac{(X-X_i)^2}{2\sigma^2}} \qquad (1)$$

such that $X \in R^2$. This function results from injecting a Parzen window into the Schrödinger wave equation [2] [3] and balancing the resulting energy. Other methods based on energy variation may also be instructive [19]. The minima of this potential provides the cluster centers for a given standard deviation σ. As stated before, we limit ourselves to two dimensions. This method is more stable and precise than the standard K-means method [3].

Moreover, and in contradistinction to other data clustering methods, the determination of the parameter σ gives a number of extrema. The number of minima is not determined beforehand but obtained numerically.

One main difficulty is to determine the minima of the potential energy. Nowadays, the technique used to approach the minima is through the gradient descent or the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithms [3]. Some investigations have been made to improve the detection of clusters via the potential energy function. For instance, in 2018, Decheng *et al.* [20] improved the quantum clustering analysis by developing a new weighted distance once a minimum had been found. Improvements are needed to capture all the minima efficiently.

The present work consists, as shown in Subsection 2.1, in simplifying the derivatives of the potential energy function such that the minima can be determined by some solution of a system of equations. Finding the extrema (minima, maxima and saddle points) of the function (1) is equivalent to solving a system

$$\begin{cases} M(x,y) = 0 \\ L(x,y) = 0 \end{cases} \qquad (2)$$

where $M(x,y)$ and $L(x,y)$ are bivariate exponential functions which can be expressed as polynomial in x, e^x, y and e^y. In this scenario, the degrees of M and L in x (respectively in y) are one. In Subsection 2.2, the implicit functions of $M = 0$ and $L = 0$ are investigated and the on going Crab example is presented Subsection 2.3. Section 3, the case $N = 2$ is formally solved and a new block approach is presented in Section 4. The aim of this new method is to reduce memory and computation costs. The main formal result is given in Subsection 4.1. We prove that the function (1) has only one minimum if the set of particles $(X_i)_{i=1..N}$ are all included in a square of side σ. Then, we propose a method based on this result and a block approach to capture all the minima in a more efficient way. The presentation of benchmarks closed Section 4. Finally, we conclude Section 5.

2. PROBLEM REDUCTION AND FIRST ANALYSIS

In this section, we transform the minimization problem of the potential energy function (1) to the resolution of a system of two equations in two variables and $2N$ parameters, namely the particles coordinates $(X_i)_{i=1..N}$.

2.1 Problem reduction

It is known that the value of σ has a crucial role on the number of minima: the greater the value of σ, the smaller the number of minima. To simplify the potential energy function, we denote $Y = \frac{X}{\sqrt{2}\sigma}$. This variable change remove σ from the function. Discussion of σ will be presented at the end of this section.

We get

$$\frac{1}{2\sigma^2}\frac{1}{\sum_{i=1}^{N} e^{-\frac{(X-X_i)^2}{2\sigma^2}}}\sum_{i=1}^{N}(X-X_i)^2 e^{-\frac{(X-X_i)^2}{2\sigma^2}} = \frac{1}{\sum_{i=1}^{N} e^{-(Y-Y_i)^2}}\sum_{i=1}^{N}(Y-Y_i)^2 e^{-(Y-Y_i)^2} \qquad (3)$$

where for all i, $Y_i = \frac{X_i}{\sqrt{2}\sigma}$. We denote this function $h(Y)$.

Theorem 1. *The extrema $Y = (x,y)$ of function*

$$h(x,y) = \frac{1}{\sum_{i=1}^{N} e^{-(Y-Y_i)^2}}\sum_{i=1}^{N}(Y-Y_i)^2 e^{-(Y-Y_i)^2} \qquad (4)$$

satisfy the system of the following two bivariate functions:

$$\begin{cases} M(x,y) = 0 \\ L(x,y) = 0 \end{cases} \qquad (5)$$

with $Y_i = (x_i, y_i)$ *for all* $i = 1..N$ *and*

$$M(x,y) = \sum_{i=1}^{N} e^{-2x_i^2-2y_i^2}\, e^{4x_i x+4y_i y}(x-x_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} e^{-x_i^2-y_i^2}\, e^{-x_j^2-y_j^2} e^{2(x_i+x_j)x+2(y_i+y_j)y} \qquad (6)$$
$$[(2x-x_i-x_j)(1-(x_i-x_j)^2) - (x_i-x_j)(y_i-y_j)(2y-y_i-y_j)]$$

$$\text{and } L(x,y) = \sum_{i=1}^{N} e^{-2x_i^2-2y_i^2}\, e^{4x_i x+4y_i y}(y-y_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} e^{-x_i^2-y_i^2}\, e^{-x_j^2-y_j^2} e^{2(x_i+x_j)x+2(y_i+y_j)y} \qquad (7)$$
$$[(2y-y_i-y_j)(1-(y_i-y_j)^2) - (y_i-y_j)(x_i-x_j)(2x-x_i-x_j)]$$

Remark: We will also use the shortest expression:

$$M(x,y) = \sum_{i=1}^{N}(x-x_i)K_i^2 + \sum_{i<j} c_{ij}\, K_i K_j \text{ and } L(x,y) = \sum_{i=1}^{N}(y-y_i)K_i^2 + \sum_{i<j} d_{ij}\, K_i K_j \qquad (8)$$

with for all i, $K_i = e^{-x_i^2-y_i^2} e^{2(x_i+x_j)x}$, and for all i, j, $i < j$, $c_{ij} = (2x-x_i-x_j)(1-(x_i-x_j)^2) - (x_i-x_j)(y_i-y_j)(2y-y_i-y_j)$ and $d_{ij} = (2y-y_i-y_j)(1-(y_i-y_j)^2) - (y_i-y_j)(x_i-x_j)(2x-x_i-x_j)$.

Proof. $h(Y)$ is a fraction of two exponential polynomials, namely $h(Y) = \frac{f(Y)}{g(Y)}$ with

$$g(Y) = \sum_{i=1}^{N} e^{-(Y-Y_i)^2} \text{ and } f(Y) = \sum_{i=1}^{N}(Y-Y_i)^2 e^{-(Y-Y_i)^2} \qquad (9)$$

Since $Y \in R^2$, Y is denoted $Y = (x, y)$, then f and g can also be written as

$$f(x,y) = \sum_{i=1}^{N}((x-x_i)^2 + (y-y_i)^2) e^{-(x-x_i)^2-(y-y_i)^2} \text{ and } g(x,y) = \sum_{i=1}^{N} e^{-(x-x_i)^2-(y-y_i)^2} \qquad (10)$$

by denoting $Y_i = (x_i, y_i)$. The extrema of $h(x,y)$ satisfy the system $\begin{cases} \frac{\partial h(x,y)}{\partial x} = 0 \\ \frac{\partial h(x,y)}{\partial y} = 0 \end{cases}$ which is equivalent to:

$\begin{cases} \frac{\partial f(x,y)}{\partial x} g(x,y) - \frac{\partial g(x,y)}{\partial x} f(x,y) = 0 \\ \frac{\partial f(x,y)}{\partial y} g(x,y) - \frac{\partial g(x,y)}{\partial y} f(x,y) = 0 \end{cases}$ since $g(x,y) \neq 0$ everywhere.

The formal computation of the equations of the last system gives expressions which can be divided by $2e^{-x^2-y^2}$. We finally obtain Theorem (1).

$$\frac{\partial f(x,y)}{\partial x}g(x,y) - \frac{\partial g(x,y)}{\partial x}f(x,y)$$
$$= 2\sum(x-x_i)e^{-(x-x_i)^2-(y-y_i)^2}[1-(x-x_i)^2-(y-y_i)^2] \times \sum e^{-(x-x_i)^2-(y-y_i)^2}$$

$$+2\sum(x-x_i)e^{-(x-x_i)^2-(y-y_i)^2} \times \sum[(x-x_i)^2+(y-y_i)^2]e^{-(x-x_i)^2-(y-y_i)^2}$$
$$= 2\sum_{i=1}^{N}(x-x_i)e^{-2(x-x_i)^2-2(y-y_i)^2} + \sum_{i=1}^{N}\sum_{j>i}^{N} 2\,c_{i,j}(x,y)e^{-(x-x_i)^2-(y-y_i)^2-(x-x_j)^2-(y-y_j)^2}.$$

We have for all i, j, $i \neq j$,

$$\begin{aligned} c_{i,j}(x,y) &= 2x - x_i - x_j + (x_i - x_j)[(x-x_i)^2 + (y-y_i)^2 - (x-x_j)^2 - (y-y_j)^2] \\ &= 2x - x_i - x_j + (x_i - x_j)(2(x_j - x_i)x + x_i^2 - x_j^2 + 2(y_j - y_i)y + y_i^2 - y_j^2) \\ &= 2x(1-(x_i-x_j)^2) - x_i - x_j + (x_i - x_j)(x_i^2 - x_j^2 + 2(y_j - y_i)y + y_i^2 - y_j^2) \\ &= 2x(1-(x_i-x_j)^2) + (x_i + x_j)((x_i-x_j)^2 - 1) + (x_i - x_j)(2(y_j - y_i)y + y_i^2 - y_j^2)) \\ &= (2x - x_i - x_j)(1-(x_i-x_j)^2) - (x_i - x_j)(y_i - y_j)(2y - y_i - y_j) \end{aligned}$$

By dividing this result by $2e^{-x^2-y^2}$, we obtain $\frac{\partial f(x,y)}{\partial x}g(x,y) - \frac{\partial g(x,y)}{\partial x}f(x,y) = 0 \Leftrightarrow M(x,y) = 0$ where $M(x,y) = \sum_{i=1}^{N}(x-x_i)K_i^2 + \sum_{i<j} c_{ij}\,K_iK_j$

and $K_i = e^{-(x-x_i)^2-(y-y_i)^2+x^2+y^2} = e^{-x_i^2-y_i^2}e^{2(x_i+x_j)x}$

Proceeding in the same way on the equation $(L2)$, we find that finding the extrema of Function (1) is equivalent to solve the system $\begin{cases} M(x,y) = \sum_{i=1}^{N}(x-x_i)K_i^2 + \sum_{i<j} c_{ij}\,K_iK_j = 0 \\ L(x,y) = \sum_{i=1}^{N}(y-y_i)K_i^2 + \sum_{i<j} d_{ij}\,K_iK_j = 0 \end{cases}$

with $d_{ij} = (2y - y_i - y_j)(1-(y_i-y_j)^2) - (y_i - y_j)(x_i - x_j)(2x - x_i - x_j)$ □

2.2 Cylindrical decomposition

For a given set of particles $(Y_i)_{i=1..N} = (x_i, y_i)_{i=1..N}$, the solutions of System (2) correspond to the intersection between the implicit functions of $M(x,y) = 0$ and those of $L(x,y) = 0$ (see Figure 1 for the example of crab with $N = 200$). Let us denote $ymax$ (resp. $xmax$) the index the greatest element of $(y_i)_{i=1..N}$ (resp. $(x_i)_{i=1..N}$) such that $\forall i \in \{1,\dots,N\} - \{ymax\}\ y_{ymax} > y_i$. In the same way, we denote $ymin$ (resp. $xmin$) the index the smallest element of $(y_i)_{i=1..N}$ (resp. $(x_i)_{i=1..N}$) such that $\forall i \in \{1,\dots,N\} - \{ymin\}\ y_{ymin} < y_i$. We have the following results:

- The infinite branches of the implicit functions of $M(x,y)$ tends to x_{ymin} at $-\infty$ and x_{ymax} at $+\infty$
- The infinite branches of the implicit functions of $L(x,y)$ tends to y_{ymin} at $-\infty$ and y_{ymax} at $+\infty$

The remainder of this subsection will display three useful lemmas, then, the proof of the stated results will be presented.

Lemma 1. *Let $q(x,y)$ be a real bivariate function in x and y which can be expressed as a polynomial in the variable y such that $q(x,y) = \sum_{i=0}^{d} a_i(x)y^i$ with $d = deg_y(q)$. Let us denote $y = \Phi(x)$ as an implicit function defined by $q(x,y) = 0$. If there exists a real x^* such that $\lim_{x\to x^*}\Phi(x) = \pm\infty$ then $a_d(x^*) = 0$.*

This lemma is an application of Cauchy's bound and some details are given in Ref. [21]. It also gives the following symmetric lemma:

Lemma 2. *Let $q(x,y)$ be a real bivariate function in x and y which can be expressed as a polynomial in the variable x such that $q(x,y) = \sum_{i=0}^{m} b_i(y)x^i$ with $m = deg_x(q)$. The finite limits at ∞ of the implicit functions of $p(x,y) = 0$ are contained in the solutions of the equation $b_m(y) = 0$.*

Finally, the third lemma is more known (see e.g. Ref. [23] for the proof) which states that

Lemma 3. $\forall s \in R^{+*}$, $\ln x \leq \frac{x^s}{s}$.

We consider the implicit functions of the equation $M(x,y) = 0$. We now prove the following theorem:

Theorem 2. *Let $(x_i)_{i=1\dots N}$ and $(y_i)_{i=1\dots N}$ be the sequences defining $M(x,y) = 0$. Assume that the greatest element of $(y_i)_{i=1\dots N}$ is reached in a value named $ymax$ such that $\forall i \in \{1,\dots,N\} - \{ymax\}$ $y_{ymax} > y_i$. Then there exists an implicit function of $M(x,y) = 0$ named $x = \Phi(y)$ such that $lim_{y\to\infty}\Phi(y) = x_{ymax}$. x_{ymax} is a finite limit at $+\infty$.*

Proof. Let us proceed to the variable changes $z = e^x$ and $t = e^y$ on $M(x,y)$. By denoting $a_i = e^{-x_i^2 - y_i^2}$, we obtain $m(z,t) = \sum_{i=1}^{N} a_i^2 z^{4x_i} t^{4y_i}(\ln(z) - x_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} a_i a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)} [(2\ln(z) - x_i - x_j)(1 - (x_i - x_j)^2) - (x_i - x_j)(y_i - y_j)(2\ln(t) - y_i - y_j)]$

$m(z,t)$ is a function of two variables z and t. Moreover, this function can also be viewed as a univariate function of the variable t with a parameter z. In that case, it will be denoted $m_z(t)$. This rule of notation holds for other functions derived from this proof.

This proof is constructed as follows: Firstly, two univariate polynomials in t named $m_z^-(t)$ and $m_z^+(t)$ are created such that $m_z^-(t) \leq m_z(t) \leq m_z^+(t)$.

Then, we will prove that $m^-(z,t) = 0$ and $m^+(z,t) = 0$ have the same finite limit in x when y tends to infinity. Finally, we prove that it is the same for $m_z(t) = 0$.

$m_z(t)$ bounds:

We define the condition C_{ij} to be $(x_i - x_j)(y_i - y_j) > 0$. Let us consider a real s such that $0 < s \leq min(|y_{ymax} - y_j|)$, and for $y > max(0, y_{ymax})$, lemma 3 gives $\max(0, y_{ymax}) \leq \ln(t) \leq \frac{t^s}{s}$.

Thus, when the condition C_{ij} holds, $2(x_i - x_j)(y_i - y_j)\max(0, y_{ymax}) \leq 2(x_i - x_j)(y_i - y_j)\ln(t) \leq 2(x_i - x_j)(y_i - y_j)\frac{t^s}{s}$, whereas when $\neg C_{ij}$ holds, $2(x_i - x_j)(y_i - y_j)\max(0, y_{ymax}) \geq 2(x_i - x_j)(y_i - y_j)\ln(t) \geq 2(x_i - x_j)(y_i - y_j)\frac{t^s}{s}$.

Thanks to the C_{ij} condition, $m_z(t)$ is decomposed into:

$$\sum_{i=1}^{N} a_i^2 z^{4x_i} t^{4y_i}(\ln(z) - x_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} a_i a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)} [(2\ln(z) - x_i - x_j)(1 - (x_i - x_j)^2) + (x_i - x_j)(y_i - y_j)(y_i + y_j)]$$
$$-2\sum_{i=1}^{N}\sum_{j>i, C_{ij}}^{N} a_i a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j)\ln(t) - 2\sum_{i=1}^{N}\sum_{j>i, \neg C_{ij}}^{N} a_i a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j)\ln(t)$$

and we then construct the polynomials $m_z^-(t)$ and $m_z^+(t)$ to be respectively: $m_z^-(t) =$

$$\sum_{i=1}^{N} a_i^2\, z^{4x_i} t^{4y_i}(\ln(z) - x_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}[(2\ln(z) - x_i - x_j)(1 - (x_i - x_j)^2) + (x_i - x_j)(y_i - y_j)(y_i + y_j)]$$

$$-2\frac{t^s}{s}\sum_{i=1}^{N}\sum_{j>i, C_{ij}}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j) - 2\max(0, y_{ymax})\sum_{i=1}^{N}\sum_{j>i, \neg C_{ij}}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j)$$

$$and\ m_z^+(t) =$$

$$\sum_{i=1}^{N} a_i^2\, z^{4x_i} t^{4y_i}(\ln(z) - x_i) + \sum_{i=1}^{N}\sum_{j>i}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}[(2\ln(z) - x_i - x_j)(1 - (x_i - x_j)^2) + (x_i - x_j)(y_i - y_j)(y_i + y_j)]$$

$$-2\max(0, y_{ymax})\sum_{i=1}^{N}\sum_{j>i, C_{ij}}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j) - 2\frac{t^s}{s}\sum_{i=1}^{N}\sum_{j>i, \neg C_{ij}}^{N} a_i\, a_j z^{2(x_i+x_j)} t^{2(y_i+y_j)}(x_i - x_j)(y_i - y_j)$$

We obtain that, if $y > \max(0, y_{ymax})$, then $m_z^-(t) < m_z(t) < m_z^+(t)$.

Finite limits: Let us denote $ymax$ the index such that $\forall i \in \{1, \dots, N\} - \{ymax\},\ y_{ymax} > y_i$.

For $y > \max(0, y_{ymax}) \Leftrightarrow t > \max(1, e^{y_{ymax}})$, $m_z^-(t)$ and $m_z^+(t)$ are constructed such that there exists at least one implicit function of $m(z,t) = 0$ between one implicit function of $m^-(z,t) = 0$ and $m^+(z,t) = 0$. Moreover $m_z^-(t)$ and $m_z^+(t)$ are polynomials having the same leading coefficient

$$a_{ymax}^2 z^{4x_{ymax}}(\ln(z) - x_{ymax}) = a_{ymax}^2 e^{4x_{ymax}x}(x - x_{ymax}).$$

Lemma 2 tells us that the finite limits at $+\infty$ of the implicit functions of $m^-(z,t)$ and $m^+(z,t)$ are contained in the solutions of the equation $a_{ymax}^2 z^{4x_{ymax}}(\ln(z) - x_{ymax}) = 0$ which is equivalent to the equation $x - x_{ymax} = 0$.

If $x > x_{ymax}$, $\lim_{t\to\infty} m_z^-(t) = +\infty$ and $\lim_{t\to\infty} m_z^+(t) = \infty$. Moreover if $x < x_{ymax}$, $\lim_{t\to\infty} m_z^-(t) = -\infty$ and $\lim_{t\to\infty} m_z^+(t) = -\infty$. Hence there exists an implicit function $x = \Phi^-(y)$ defined by $m^-(z,t) = 0$ and there exists an implicit function $z = \Phi^+(t)$ defined by $m^+(z,t) = 0$ such that $lim_{t\to\infty}\Phi^-(t) = e^{x_{ymax}}$ and $lim_{t\to\infty}\Phi^-(t) = e^{x_{ymax}}$. We conclude that $lim_{t\to\infty}\Phi(t) = e^{x_{ymax}}$ and $M(x,y) = 0$ admit x_{ymax} has a finite limit at infinity.

Remark: If (y_i) are not integers but are instead rational, it is easy to find λ such that $\forall i \in [1, N], \lambda y_i \in N\ and\ \lambda s \in N$, $m_z^-(t)$ and $m_z^+(t)$ can be easily transformed into polynomials $M_z^-(T)$ and $M_z^+(T)$ by applying $t = T^\lambda$. Both $M_z^-(T)$ and $M_z^+(T)$ have the same leading coefficient $a_{ymax}^2 z^{4x_{ymax}}(\ln(z) - x_{ymax}) = a_{ymax}^2 e^{4x_{ymax}x}(x - x_{ymax})$. By means of lemma 2, the finite limits at $+\infty$ of the implicit functions are contained in the solutions of the equation $x - x_{ymax} = 0$. if $x > x_{ymax}$, $\lim_{T\to\infty} M_z^-(T) = \lim_{t\to\infty} m_z^-(t) = +\infty$ and $\lim_{T\to\infty} M_z^+(T) = \lim_{t\to\infty} m_z^+(t) = \infty$. Moreover, if $x < x_{ymax}$, $\lim_{T\to\infty} M_z^-(T) = \lim_{t\to\infty} m_z^-(t) = -\infty$ and $\lim_{T\to\infty} M_z^+(T) = \lim_{t\to\infty} m_z^+(t) = -\infty$. □

Since for large z and large t, $m(z,t) > m^-(z,t) > 0$ we obtain the following theorem:

Theorem 3. *$M(x,y) = 0$ has no infinite branches at infinity.*

From Theorem 2 and Theorem 3, we derived that in the half-plane $t > e^{y_{ymax}}$, there is only one implicit function which is living between $m^-(z,t) = 0$ and $m^+(z,t) = 0$.

Theorem 4. *Let* $(x_i)_{i=1\ldots N}$ *and* $(y_i)_{i=1\ldots N}$ *the sequences defining* $M(x,y)=0$. *Assume that the smallest element of* $(y_i)_{i=1\ldots N}$ *is reached in an index value named ymin such that* $\forall i \in \{1,\ldots,N\}-\{ymin\}\ y_{ymin} < y_i$. *Then there exists an implicit function of* $M(x,y)=0$ *named* $x=\Phi(y)$ *such that* $lim_{y\to-\infty}\Phi(y) = x_{ymin}$. x_{ymin} *is a finite limit at* $-\infty$.

Theorem 4 can be proven in a manner similar to that of theorem 2 with the variable change $t=e^{-y}$.

Equivalent results can be obtained for the equation $L(x,y)=0$

Theorem 5. *Let* $(x_i)_{i=1\ldots N}$ *and* $(y_i)_{i=1\ldots N}$ *the sequences defining* $L(x,y)=0$. *Assume that the greatest element of* $(x_i)_{i=1\ldots N}$ *is reached in a index named xmax such that* $\forall i \in \{1,\ldots,N\}-\{xmax\}\ x_{xmax} > x_i$. *Then there exists an implicit function of* $L(x,y)=0$ *named* $y=\Phi(x)$ *such that* $lim_{yx\to\infty}\Phi(x) = y_{xmax}$.

Theorem 6. *Let* $(x_i)_{i=1\ldots N}$ *and* $(y_i)_{i=1\ldots N}$ *the sequences defining* $L(x,y)=0$. *Assume that the smallest element of* $(x_i)_{i=1\ldots N}$ *is reached in a index named xmin such that* $\forall i \in \{1,\ldots,N\}-\{xmin\}\ x_{xmin} < x_i$. *Then there exists an implicit function of* $L(x,y)=0$ *named* $y=\Phi(x)$ *such that* $lim_{x\to-\infty}\Phi(x) = y_{xmin}$. y_{xmin} *is a finite limit at* $-\infty$.

Again $L(x,y)=0$ has no infinite branches at infinity.

2.3 Crab example

To illustrate our results, we use the crab data clustering example [3] using the dataset from Refs. [24] [25]. This two dimensional case has been presented in Refs. [2] [3]. This example is composed of four classes at 50 samples each, making a total of 200 samples i.e. particles and by taking $\sigma = 0.05$, we obtain, after the variable change described in Section 2, a set of particles for which the x and y coordinates $(x_i)_{i=1..200}$ and $(y_i)_{i=1..200}$ satisfy $xmin = 150$, $xmax = 65$, $y_{xmax} = -0.3190$, $y_{xmin} = 0.3640$, $ymin = 35$, $ymax = 105$, $x_{ymax} = 0.0038$,$x_{ymin} = -0.7941$. The curve $M(x,y)=0$ is shown, Figure 1 (a) and (b), in red and the curve $L(x,y)=0$ is shown in green. The intersection between the red and the green curves corresponds to the extrema of h.

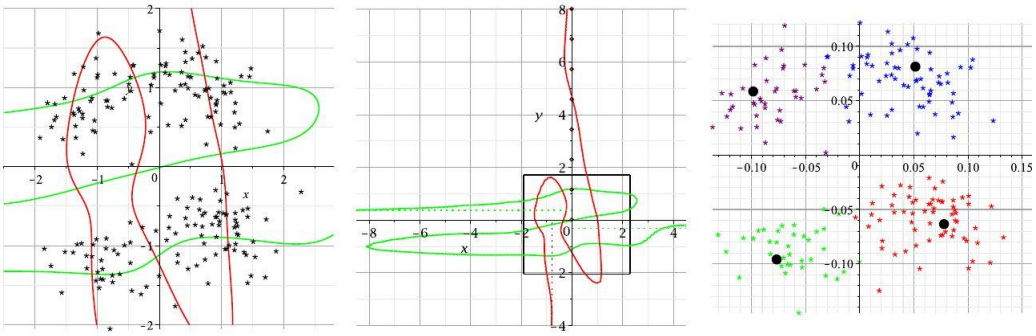

Figure 1: Crab example with $\sigma = 0.05$*: (a) the set of points* $(x,y)_i$*, and the implicit curves of* $M(x,y)=0$ *and* $L(x,y)=0$. *(b) the limits of implicit curves.(c) Crab clusters produced by using the minimum Euclidean distance from the minima (*$\sigma = 0.05$*)*

Using the Maple computer algebra system [26], we obtain one maximum, four minima and four saddle points. The clusters produced by using the minimum Euclidean distance from the minima are shown Figure 1 (c). For larger σ, the number of solutions decreases and hence, a coarser clustering is found. A deeper analysis is provided by Table 1. It gives for some σ ranges the resulting number of clusters. It shows that the non trivial number of clusters is more likely 4 because the corresponding σ range is the widest.

σ range	$0.085 \leq$	$[0.074, 0.084]$	$[0.071, 0.073]$	$[0.025, 0.070]$	$0.02 \geq$
clusters number	1	2	3	4	≥ 5

Table 1: Range of σ with respect to number of clusters.

This first example of 200 samples can be fully solved numerically but the corresponding function $M(x, y)$ and $L(x, y)$ are sums of 20100 monomials in x, y, e^x and e^y. The size of M and L is an issue and the aim of the following section is to reduce the size of M and L while maintaining a good approximation of minima.

3. THE CASE N=2

In this section we present the case where $N = 2$. The potential energy function becomes

$$\frac{1}{2\sigma^2}\frac{1}{e^{-\frac{(X-X_1)^2}{2\sigma^2}}+e^{-\frac{(X-X_2)^2}{2\sigma^2}}}\left((X-X_1)^2e^{-\frac{(X-X_1)^2}{2\sigma^2}}+(X-X_2)^2e^{-\frac{(X-X_2)^2}{2\sigma^2}}\right) \qquad (11)$$

The minimization problem is then reduced to the resolution of the System (12) defined by

$$\begin{cases} a_1^2e^{4x_1x+4y_1y}(x-x_1)+a_2^2e^{4x_2x+4y_2y}(x-x_2) \\ +a_1a_2e^{2(x_1+x_2)x+2(y_1+y_2)y}[(2x-x_1-x_2)(1-(x_1-x_2)^2)-(x_1-x_2)(y_1-y_2)(2y-y_1-y_2)]=0 \\ a_1^2e^{4x_1x+4y_1y}(y-y_1)+a_2^2e^{4x_2x+4y_2y}(y-y_2) \\ +a_1a_2e^{2(x_1+x_2)x+2(y_1+y_2)y}[(2y-y_1-y_2)(1-(y_1-y_2)^2)-(x_1-x_2)(y_1-y_2)(2x-x_1-x_2)]=0 \end{cases} \qquad (12)$$

where $a_1 = e^{-x_1^2-y_1^2}$ and $a_2 = e^{-x_2^2-y_2^2}$ are constants depending on the coordinates of the particles.

In the following, we prove that

- If $(X_1 - X_2)^2 \leq 2\sigma^2$, Function (11) has one and only one minimum which is $\frac{X_1+X_2}{2\sqrt{2}\sigma}$;
- Else, Function (11) has two minima and we give a way to compute them easily.

We begin with a useful lemma dealing with the function $e^x = -\frac{x+w}{x-w}$. It belongs to the class of offset logarithm functions $e^x\frac{x-w}{x+w} = a$ [27].

Lemma 4. *The function $e^x = -\frac{x+w}{x-w}$ has only one solution $x = 0$ when $w \leq 2$. If $w > 2$, it has three solutions including $x = 0$.*

Proof. We use the derivative of $h(x) = e^x(x-w)+x+w$. $\frac{dh}{dx} = e^x(x-w+1)+1$ has no solution when $w < 2$. It has 1 solution which is $x = 0$ if $w = 2$. if $w > 2$ the two solutions of $\frac{dh}{dx} = 0$ are $W_0(-exp(-w+1))+w-1$ and $W_{-1}(-exp(-w+1))+w-1$, where W is the Lambert W function [28], [29], [30], [31]. Finally, we directly obtain the lemma. □

The $N = 2$ case brings us back to the standard Lambert W function with applications in symmetry in the context of linear molecules namely the double-well Dirac potential problem [32], [33]. The latter provides a one-dimensional model of the diatomic molecule known as the Hydrogen molecular ion whose eigenenergies in its three-dimensional version are also in terms of a fully generalized Lambert Function [34]. It is also seen in quantum gravity [35], [36] and in semiconductor devices [37] found in applications of Solid-state Physics. The ubiquity of applications is not surprising as they all involve exponential polynomials of two physical bodies. Not surprisingly, the Lambert W function describes the nodal structure

of the lowest discrete energy states of the two-electron system, namely the Helium atom [8] and seems ubiquitous to nature.

Theorem 7. *System* (12) *has one solution if* $1 \geq (x_1 - x_2)^2 + (y_1 - y_2)^2$ *and the solution is* $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$. *If* $1 < (x_1 - x_2)^2 + (y_1 - y_2)^2$, S_2 *has three solutions. One of them is* $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$. *The others are the* (x, y) *solutions such that* y *satisfies the univariate equations* $e^{2\frac{y}{y_1-y_2}u-u}(y - y_1) + e^{-2\frac{y}{y_1-y_2}u+u}(y - y_2) + 2(y - \frac{y_1+y_2}{2})(1 - u) = 0$ *with* $u = (x_1 - x_2)^2 + (y_1 - y_2)^2$ *and* $x = (y - \frac{y_1+y_2}{2})\frac{x_1-x_2}{y_1-y_2} + \frac{x_1+x_2}{2}$.

Proof. We first centralize the system by setting $\alpha = x - \frac{x_1+x_2}{2}$ and $\beta = y - \frac{y_1+y_2}{2}$. Then System (12) is equivalent to the system:

$$\begin{cases} e^{4x_1\alpha+4y_1\beta} & (\alpha - \frac{x_1 - x_2}{2}) + e^{4x_2\alpha+4y_2\beta}(\alpha + \frac{x_1 - x_2}{2}) \\ & +e^{2(x_1+x_2)\alpha+2(y_1+y_2)\beta}[2\alpha(1 - (x_1 - x_2)^2) - 2\beta(x_1 - x_2)(y_1 - y_2)] = 0 \\ e^{4x_1\alpha+4y_1\beta} & (\beta - \frac{y_1 - y_2}{2}) + e^{4x_2\alpha+4y_2\beta}(\beta + \frac{y_1 - y_2}{2}) \\ & +e^{2(x_1+x_2)\alpha+2(y_1+y_2)\beta}[2\beta(1 - (y_1 - y_2)^2) - 2\alpha(x_1 - x_2)(y_1 - y_2)] = 0 \end{cases}$$

$\Leftrightarrow$

$$\begin{cases} e^{2(x_1-x_2)\alpha+2(y_1-y_2)\beta} & \left(\alpha - \frac{x_1-x_2}{2}\right) + e^{-2(x_1-x_2)\alpha-2(y_1-y_2)\beta}\left(\alpha + \frac{x_1-x_2}{2}\right) + 2\alpha(1 - (x_1 - x_2)^2) - 2\beta(x_1 - x_2)(y_1 - y_2) = 0 \\ e^{2(x_1-x_2)\alpha+2(y_1-y_2)\beta} & (\beta - \frac{y_1-y_2}{2}) + e^{-2(x_1-x_2)\alpha-2(y_1-y_2)\beta}(\beta + \frac{y_1-y_2}{2}) + 2\beta(1 - (y_1 - y_2)^2) - 2\alpha(x_1 - x_2)(y_1 - y_2) = 0 \end{cases}$$

Since the point $(\alpha, \beta) = (0,0)$ is a solution of the system, $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$ is a solution of the System (12). Moreover, some linear combinations simplify System (12). Let us denote $(L1)$ and $(L2)$ as respectively the first and second lines of System (12).

$(L1)(\beta + \frac{y_1-y_2}{2}) - (L2)(\alpha + \frac{x_1-x_2}{2})$ gives $(\beta(x_1 - x_2) - \alpha(y_1 - y_2))(e^{2(x_1-x_2)\alpha+2(y_1-y_2)\beta} + 2\alpha(x_1 - x_2) + 2\beta(y_1 - y_2) + 1) = 0$ and $(L1)(\beta - \frac{y_1-y_2}{2}) - (L2)(\alpha - \frac{x_1-x_2}{2})$ gives $(\beta(x_1 - x_2) - \alpha(y_1 - y_2))(e^{-2(x_1-x_2)\alpha-2(y_1-y_2)\beta} - 2\alpha(x_1 - x_2) - 2\beta(y_1 - y_2) + 1) = 0$

Since $e^{-2(x_1-x_2)\alpha-2(y_1-y_2)\beta} - 2\alpha(x_1 - x_2) - 2\beta(y_1 - y_2) + 1$ and $e^{2(x_1-x_2)\alpha+2(y_1-y_2)\beta} + 2\alpha(x_1 - x_2) + 2\beta(y_1 - y_2) + 1$ have no common root, the System (12) is equivalent to the System (13) defined by

$$\begin{cases} e^{2(x_1-x_2)\alpha+2(y_1-y_2)\beta} & (\beta - \frac{y_1 - y_2}{2}) + e^{-2(x_1-x_2)\alpha-2(y_1-y_2)\beta}(\beta + \frac{y_1 - y_2}{2}) \\ & +2\beta(1 - (y_1 - y_2)^2) - 2\alpha(x_1 - x_2)(y_1 - y_2) = 0 \\ \beta(x_1 - x_2) - \alpha(y_1 - & y_2) = 0 \end{cases} \quad (13)$$

Finally, System (12) is equivalent to the simplified system:

$$\begin{cases} e^{2\beta\frac{(x_1-x_2)^2+(y_1-y_2)^2}{y_1-y_2}}(\beta-\frac{y_1-y_2}{2})+e^{-2\beta\frac{(x_1-x_2)^2+(y_1-y_2)^2}{y_1-y_2}}(\beta+\frac{y_1-y_2}{2})+2\beta(1-(x_1-x_2)^2-(y_1-y_2)^2)=0 \\ \alpha=\beta\frac{x_1-x_2}{y_1-y_2} \end{cases} \quad (14)$$

The first equation of this last system depends only on the variable β. By denoting $u=(x_1-x_2)^2+(y_1-y_2)^2$ and $v=y_1-y_2$, it can be rewritten as

$$h(\beta)=e^{2\beta\frac{u}{v}}\left(\beta-\frac{v}{2}\right)+e^{-2\beta\frac{u}{v}}\left(\beta+\frac{v}{2}\right)+2\beta(1-u) \quad (15)$$

Assume $v>0$ without loss of generality, h is a C^∞ function such that $\frac{dh}{d\beta}(\beta)=2\frac{u}{v}\beta(e^{2\beta\frac{u}{v}}-e^{-2\beta\frac{u}{v}})+(1-u)(e^{2\beta\frac{u}{v}}+e^{-2\beta\frac{u}{v}}+2)$ and there limits are $\lim_{\beta\to+\infty}h(\beta)=+\infty$ and $\lim_{\beta\to-\infty}h(\beta)=-\infty$. Its number of solutions depends on the sign of $1-u$.

If $1-u\geq 0$, $\frac{dh}{d\beta}(\beta)>0$ and the only solution of h is 0. This solution corresponds to a minimum of the initial problem.

If $1-u<0$, we have $h(0)=0$, $\frac{dh}{d\beta}(0)=4(1-u)<0$, $\lim_{\beta\to+\infty}\frac{dh}{d\beta}(\beta)>0$ and $\lim_{\beta\to-\infty}\frac{dh}{d\beta}(\beta)>0$.Thus h has at least 3 solutions. The solution 0 corresponds to a local maximum of the initial problem. Let us prove that h has exactly 3 solutions. Since the number of solution of $\frac{d^2h}{d^2\beta}(\beta)=\frac{2u}{v}e^{-2u\beta/v}(e^{4u\beta/v}(\frac{2u}{v}\beta-u+2)+\frac{2u}{v}\beta+u-2)$ is equal to that of $\frac{d^2h^*}{d^2\beta}(\beta)=e^{4u\beta/v}(\frac{2u}{v}\beta-u+2)+\frac{2u}{v}\beta+u-2$. $\frac{d^2h^*}{d^2\beta}=0$ is of the form $e^x=-\frac{x+w}{x-w}$ by setting $x=\frac{4u\beta}{v}$ and $w=2(u-2)$. Thanks to Lemma 4, it has at most 3 solutions. Moreover, in both cases $\frac{dh}{d\beta}(0)<0$, $\lim_{\beta\to+\infty}\frac{dh}{d\beta}(\beta)>0$ and $\lim_{\beta\to-\infty}\frac{dh}{d\beta}(\beta)>0$ meaning that $\frac{dh}{d\beta}$ has exactly two solutions (symmetric) and h has three solutions.

□

4. THE BLOCK APPROACH

In this section, we present a new numerical approach per block. First we present the algebraic property needed to develop the new algorithm presented theoretically in the second subsection and algorithmically in the third subsection. Finally the Crab example is revisited and some other benchmarks are presented.

4.1 σ estimations

We have seen (Table 1) that the σ value is of crucial importance to the number of minima. The greater σ is, the smaller the number of minima. But obviously the number of minima also depends on the data. In this subsection, we link the value of σ with the values of the initial data in order to obtain a bound from which the number of minima is one.

Theorem 8. *Consider a set of particles $(X_i)_{i=1..N}$ where for all $i=1..N$, $X_i=(v_i,w_i)$, the potential energy function $\frac{1}{2\sigma^2}\frac{1}{\sum_{i=1}^N e^{-\frac{(X-X_i)^2}{2\sigma^2}}}\sum_{i=1}^N(X-X_i)^2e^{-\frac{(X-X_i)^2}{2\sigma^2}}$ has only one minimum for*

$$\sigma=max(v_{max}-v_{min},w_{max}-w_{min}). \quad (16)$$

Proof. To complete this proof, we prove the equivalent property: System $(2)\begin{cases}M(x,y)=0\\L(x,y)=0\end{cases}$ has only one solution if the set of points $(x_i,y_i)_{i=1..N}$ lies in a square of side $\frac{1}{\sqrt{2}}$.

First, we define the center of the square which is $C=(x_c,y_c)$ with $x_c=\frac{x_{max}+x_{min}}{2}$ and $y_c=\frac{y_{max}+y_{min}}{2}$ and proceed to the variable changes $\alpha=x-x_c$ and $\beta=y-yc$, $\alpha_i=x_i-x_c$, $\beta_i=y_i-y_c$ for all $i=1\dots N$. We obtain $M(x,y)=e^{-2x_c^2-2y_c^2+4x_cx+4y_cy}M_C(\alpha,\beta)$ where $M_C(\alpha,\beta)$ is defined by the sequences $(\alpha_i)_{i=1..N}$ and $(\beta_i)_{i=1..N}$ such that

$$M_C(\alpha,\beta)=\sum_{i=1}^{N}e^{-2\alpha_i^2-2\beta_i^2}\,e^{4\alpha_i\alpha+4\beta_i\beta}(\alpha-\alpha_i)+\sum_{i=1}^{N}\sum_{j>i}^{N}e^{-\alpha_i^2-\beta_i^2}\,e^{-\alpha_j^2-\beta_j^2}e^{2(\alpha_i+\alpha_j)\alpha+2(\beta_i+\beta_j)\beta}c_{ij}. \quad (17)$$

Where $c_{ij}=(2\alpha-\alpha_i-\alpha_j)(1-(\alpha_i-\alpha_j)^2)-(\alpha_i-\alpha_j)(\beta_i-\beta_j)(2\beta-\beta_i-\beta_j)$. In the same way, $L(x,y)=e^{-2x_c^2-2y_c^2+4x_cx+4y_cy}L_C(\alpha,\beta)$ with

$$L_C(\alpha,\beta)=\sum_{i=1}^{N}e^{-2\alpha_i^2-2\beta_i^2}\,e^{4\alpha_i\alpha+4\beta_i\beta}(\beta-\beta_i)+\sum_{i=1}^{N}\sum_{j>i}^{N}e^{-\alpha_i^2-\beta_i^2}\,e^{-\alpha_j^2-\beta_j^2}e^{2(\alpha_i+\alpha_j)\alpha+2(\beta_i+\beta_j)\beta}d_{ij}. \quad (18)$$

where $d_{ij}=(2\beta-\beta_i-\beta_j)(1-(\beta_i-\beta_j)^2)-(\alpha_i-\alpha_j)(\beta_i-\beta_j)(2\alpha-\alpha_i-\alpha_j)$.

Now we have to prove the equivalent statement: the system $\begin{cases}M_C(\alpha,\beta)=0\\L_C(\alpha,\beta)=0\end{cases}$ has only one solution if for all $i\in[1,N]$, $|\alpha_i|\le\frac{1}{2\sqrt{2}}$ and $|\beta_i|\le\frac{1}{2\sqrt{2}}$.

The proof of the last statement is decomposed into two parts. First, we prove that there is at most one minimum: $\frac{dM_C}{d\alpha}(\alpha,\beta)$ is composed of a sum of terms of the form $e^{-2\alpha_i^2-2\beta_i^2}e^{4\alpha_i\alpha+4\beta_i\beta}(4\alpha_i(\alpha-\alpha_i)+1)$ and $e^{-\alpha_i^2-\beta_i^2}e^{-\alpha_j^2-\beta_j^2}e^{2(\alpha_i+\alpha_j)\alpha+2(\beta_i+\beta_j)\beta}(2(\alpha_i+\alpha_j)[(2\alpha-\alpha_i-\alpha_j)(1-(\alpha_i-\alpha_j)^2)-(2\beta-\beta_i-\beta_j)(\alpha_i-\alpha_j)(\beta_i-\beta_j)]+2(1-(\alpha_i-\alpha_j)^2))$. All those terms are positive when $\alpha_i,\beta_i,\alpha,\beta\in[-\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}}]$ thus $\frac{dM_C}{d\alpha}(\alpha,\beta)>0$. In the same way $\frac{dL_C}{d\beta}(\alpha,\beta)>0$. Here again, in the phase space, there is at most one solution in α of $M_C(\alpha,\bar{\beta})$ for all $\bar{\beta}\in R$ and $\alpha\in[-\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}}]$ and there is at most one solution in y of $L_C(\bar{\alpha},\beta)$ for all $\bar{\alpha}\in R$ and $\beta\in[-\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}}]$. $\frac{dM_C(\alpha,\beta}{d\alpha}>\frac{dM_{1,2}(\alpha,\beta}{d\alpha}>0$ and $\frac{dL_C}{d\beta}(\alpha,\beta)>\frac{dL_{1,2}}{d\beta}(\alpha,\beta)>0$ meaning that there is at most one solution in $[-\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}}]$.

Secondly, we prove that at least one implicit curve of $M=0$ lies in the square $[-\frac{1}{2\sqrt{2}},\frac{1}{2\sqrt{2}}]^2$. To do it, it is enough to prove that $M_C(\alpha,\beta)>0$ for $\alpha>\frac{1}{2\sqrt{2}}$ and $M_C(\alpha,\beta)<0$ for $\alpha<-\frac{1}{2\sqrt{2}}$: Without loss of generality, we set $(\alpha_i)_{i=1..N}$ which satisfies for all $j>i$, $\alpha_i>\alpha_j$. A simple index permutation allows us to order the sequence $(\alpha_i)_{i=1..N}$.

We know that for all $j>i$, $c_{i,j}(\alpha,\beta)=2\alpha-\alpha_i-\alpha_j+(\alpha_i-\alpha_j)[(\alpha-\alpha_i)^2+(\beta-\beta_i)^2-(\alpha-\alpha_j)^2-(\beta-\beta_j)^2]$. From the triangular inequality, we obtain $2\alpha-\alpha_i-\alpha_j-|\alpha_i-\alpha_j|[(\alpha_i-\alpha_j)^2+(\beta_i-\beta_j)^2]\le c_{i,j}(\alpha,\beta)$ and $c_{i,j}(\alpha,\beta)\le 2\alpha-\alpha_i-\alpha_j+|\alpha_i-\alpha_j|[(\alpha_i-\alpha_j)^2+(\beta_i-\beta_j)^2]$Since $(\alpha_i-\alpha_j)^2+(\beta_i-\beta_j)^2\le 1$, we get $c_{i,j}(\alpha,\beta)\ge 2\alpha-\alpha_i-\alpha_j-(\alpha_i-\alpha_j)=2\alpha-2\alpha_i$ and $c_{i,j}(\alpha,\beta)\ge 0$. Finally $M_C(\alpha,\beta)>0$ for all $\alpha>\frac{1}{2\sqrt{2}}$. With the same approach, we get

- $M_C(\alpha,\beta) < 0$ for all $\alpha < -\frac{1}{2\sqrt{2}}$, $\beta \in R$
- $L_C(\alpha,\beta) > 0$ for all $\beta > \frac{1}{2\sqrt{2}}$, $\alpha \in R$
- $L_C(\alpha,\beta) < 0$ for all $\beta < -\frac{1}{2\sqrt{2}}$, $\alpha \in R$

Finally there is one implicit function of $M_C(\alpha,\beta) = 0$ for all $\beta \in R$ when $\alpha \in [-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}]$ and there is one implicit function of $L_C(\alpha,\beta) = 0$ for $\alpha \in R$ when $\beta \in [-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}]$. We conclude that the two curves intersect in the square $[-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}]^2$ and this intersection corresponds to a minimum. □

For instance, in the crab example, $max(v_{max} - v_{min}, w_{max} - w_{min}) = 0.297$ and without any computation, we know that if $\sigma \geq 0.297$, the function (1) has exactly one minimum.

To serve our new block method presented next subsection, we give another formulation of Theorem 2 as a corollary

Corollary 1. *The bivariate function* $\frac{1}{2\sigma^2}\frac{1}{\sum_{i=1}^{N} e^{-\frac{(X-X_i)^2}{2\sigma^2}}}\sum_{i=1}^{N}(X-X_i)^2 e^{-\frac{(X-X_i)^2}{2\sigma^2}}$ *has only one minimum if the set of points* $(X_i)_{i=1..N}$ *are all included in a square of side* σ.

4.2 System approximation construction

In the general case of N particles, the functions $M(x,y)$ and $L(x,y)$ are sums of $\frac{N(N+1)}{2}$ exponential polynomials of the form $(x-x_i)K_i^2$, $c_{ij}K_iK_j$ or $d_{ij}K_iK_j$. We recall System (2): $\begin{cases} M(x,y) = 0 \\ L(x,y) = 0 \end{cases}$

such that

$$M(x,y) = \sum_{i=1}^{N}(x-x_i)K_i^2 + \sum_{i<j} c_{ij}\,K_iK_j \quad and \quad L(x,y) = \sum_{i=1}^{N}(y-y_i)K_i^2 + \sum_{i<j} d_{ij}\,K_iK_j \qquad (19)$$

where $K_i = e^{-(x-x_i)^2-(y-y_i)^2+x^2+y^2}$.

When N is large, we need a strategy to decrease the length of $M(x,y)$ and $L(x,y)$ while maintaining the main property of System (2) which is to define the cluster centers.

Let us denote $R = [x_{min}, x_{max}] \times [y_{min}, y_{max}]$ the rectangle containing all the points $(Y_i)_{i=1..N}$. The basic idea is to partition R into squares and approximate the minimum locally by considering for each square, only its particles. These new points will correspond to a weighted approximation of the particles in the square. They will therefore correspond to the weighted particles of the approximate system.

The block construction consists of subdividing R into k^2 square blocks of length $\frac{1}{k}max(x_{max} - x_{min}, y_{max} - y_{min})$. Since the particles are numbered from 1 to N, we denote $B(i)$ the block containing the particle i. i is named a representative of the block and we have: $B(i) = B(j)$ if i and j belong to the same square. We denote R a set containing exactly one representative of each non empty block.

Let $\alpha \in R$, the function M is reduced to the particles of the block $B(\alpha)$ which is denoted $M_{B(\alpha)}$ and

$$M_{B(\alpha)}(x,y) = \sum_{i\in B(\alpha)}(x-x_i)K_i^2 + \sum_{i<j;\, i\in B(\alpha), j\in B(\alpha)} c_{ij}\,K_iK_j \qquad (20)$$

$$Similarly, \qquad L_{B(\alpha)}(x,y) = \sum_{i\in B(\alpha)} (y-y_i)K_i^2 + \sum_{i<j, i\in B(\alpha), j\in B(\alpha)} d_{ij}\, K_i K_j \qquad (21)$$

By setting $\sigma = \frac{1}{k} max(x_{max} - x_{min}, y_{max} - y_{min})$, theorem 8 guarantees that the system governed by $\begin{cases} M_{B(\alpha)}(x,y) = 0 \\ L_{B(\alpha)}(x,y) = 0 \end{cases}$ has exactly one minimum $(x_{B(\alpha)}, y_{B(\alpha)})$.

Therefore, $M(x,y) = \sum_{\alpha\in R} M_{B(\alpha)} + \sum_{i<j, j\notin B(i)} c_{ij}\, K_i K_j$ and we approximate $M(x,y)$ by

$$M_{Bls}(x,y) = \sum_{\alpha} p_{B(\alpha)} \left(x - x_{B(\alpha)}\right) K_{B(\alpha)}^2 + \sum_{k\in R, l\in R, \alpha<\beta} p_{B(\alpha)}\, p_{B(\beta)} c_{B(\alpha)B(\beta)} K_{B(\alpha)} K_{B(\beta)} \qquad (22)$$

where $p_{B(\alpha)}$ corresponds to the number of particles inside $B(\alpha)$. Equivalently, we approximate $L(x,y)$ by $L_{Bls}(x,y)$ to obtain the block-system $\begin{cases} M_{Bls} = 0 \\ L_{Bls} = 0 \end{cases}$. M_{Bls} and L_{Bls} are now sums of at most $\frac{k^2(k^2+1)}{2}$ exponential polynomials and $k^2 << N$.

Remark (Limit preservation): the minima of System (2) are usually in the domain R. Nevertheless, the limit preservation of the approximate system is important. To do so, and according to Section 2, the four extrema (x_{minx}, y_{minx}), (x_{miny}, y_{miny}), (x_{maxx}, y_{maxx}) and (x_{maxy}, y_{maxy}) are usually not integrated into blocks and appear without any modification in the block-system.

4.3. Algorithm

The main steps of the algorithm are as follows:

- Input: the list of particles $(X_i)_{i=1..N} = ((v_i, w_i))_{i=1..N}$ and the parameter k of the block partition.
- Compute $\sigma = \frac{1}{k} max(v_{max} - v_{min}, w_{max} - w_{min})$.
- Normalize $(X_i)_{i=1..N}$ by setting $\forall i \in \{1..N\}, Y_i = \frac{X_i}{\sqrt{2}\sigma}$.
- Set $L = (Y_i)_{i=1..N} = ((x_i, y_i))_{i=1..N}$.
- For all $(\alpha, \beta) \in \{1..k\}^2$
  - Compute $B = [x_{min} + \alpha\sigma, x_{min} + (\alpha+1)\sigma] \times [y_{min} + \beta\sigma, y_{min} + (\beta+1)\sigma]$.
  - Find the list L_B of all the particles of L belonging to B.
  - If $L_B \neq \emptyset$ compute the minimum m_B of the block-system $\begin{cases} M_B = 0 \\ L_B = 0 \end{cases}$ involving only the particles of L_B.
  - The weight p_B of this minimum corresponds to the number of particles inside the square. $p_B = card(L_B)$.
- Consider the list L_m of all the minima with their corresponding weight. Compute the minima of the corresponding block system $\begin{cases} M_{Bls} = 0 \\ L_{Bls} = 0 \end{cases}$ involving L_m.
- Deduce the cluster centers in the $X-$variable.

Remarks:

1-With regards to the third item: If needed, one can also centralize the data by setting $C = (\frac{v_{min}+v_{max}}{2}, \frac{w_{min}+w_{max}}{2})$ and $\forall i \in \{1..N\}, Y_i = \frac{X_i - C}{\sqrt{2}\sigma}$. The centralization and normalization of the data is very important to avoid certain computational issues.

2-With regards to the fifth item: We have proven, thanks to Corollary 1, that $\begin{cases} M_B = 0 \\ L_B = 0 \end{cases}$ has exactly one minimum m_B. Indeed, the size of the block B is σ and the construction of the function M_B and L_B involves only the particles in the block B. This minimum is often close to the mass center of the cluster. Finding this minimum using a Newton-Raphson method with the mass center as a starting point has fast convergence. Moreover, one can consider a variation of our approach where σ depends on an additional parameter $l \geq 1$: $\sigma = \frac{l}{k} max(v_{max} - v_{min}, w_{max} - w_{min})$. In this variation, theorem 8 holds since $l \geq 1$ and σ and k can be chosen independently such that $\frac{\sigma k}{max(v_{max} - v_{min}, w_{max} - w_{min})} \geq 1$. Therefore we can consider an approximation involving more blocks without changing σ.

4.4 Crab Example Revisited

The block algorithm has been tested on the crab example [3] [24] [25] with varying values of k. For $k = 5$, we have reduced the minimizing problem on 200 particles to a minimizing problem on 23 weighted particles. These new 23 particles correspond to minima of a sub-problem reduced to blocks. Table 2 shows for various k, the number of non empty blocks it produces (column two) and the value of $\sigma = \frac{1}{k} max(x_{max} - x_{min}, y_{max} - y_{min})$ (column 3). It also shows, in the fourth column, the approximation of the minima of the block-system in the X variable, whereas the sixth column shows the approximation of the minima of the original System (2). In the fifth and seventh column, the number of particles per clusters is given.

k *and* σ	blocks numb.	minima $\begin{cases} \mathcal{M}_{Bls} = 0 \\ \mathcal{L}_{Bls} = 0 \end{cases}$	clust. size	minima $\begin{cases} \mathcal{M} = 0 \\ \mathcal{L} = 0 \end{cases}$	clust. size
5	23	$[-0.102270, 0.0658670]$	40	$[-0.09612, 0.06528]$	41
and		$[-0.083935, -0.103749]$	36	$[-0.07728, -0.10097]$	36
0.0594		$[0.0474319, 0.0895055]$	60	$[0.04818, 0.08389]$	59
		$[0.084879, -0.073041]$	64	$[0.07861, -0.06591]$	64
6	30	$[-0.104058, 0.0584710]$	41	$[-0.09830, 0.05817]$	41
and		$[-0.080777, -0.097700]$	36	$[-0.07653, -0.09568]$	36
0.0495		$[0.0540185, 0.0816837]$	59	$[0.05145, 0.08114]$	59
		$[0.078933, -0.065498]$	64	$[0.07766, -0.06330]$	64
9	53	$[-0.100231, 0.0463999]$	41	$[-0.09671, 0.04727]$	41
and		$[-0.072107, -0.086702]$	37	$[-0.07653, -0.08632]$	37
0.0330		$[0.062279, 0.0685270]$	59	$[0.06196, 0.06852]$	59
		$[0.076056, -0.054503]$	63	$[0.07562, -0.05440]$	63

Table 2: Comparison of the minima and the clusters using the block method and the direct method.

The clusters are obtained by computing the Euclidean distance between a particle and the four minima namely m_1, m_2, m_3 and m_4. A particle p belongs to the cluster i if $|pm_i| = min(|pm_1|, |pm_2|, |pm_3|, |pm_4|)$.
We have compared the clusters produced by the direct method with $\sigma = 5$ and those produced by the block method with $k = 5$, we observe that the result is the same except for one particle. For $k = 6$ or $k = 9$, we obtain the same clusters from both methods.

4.5 Benchmarks

The block method can be tested on larger set of particles. In this subsection, we propose three other examples (A fourth one dealing with exoplanet [38] is presented in [39]):

Gionis *et al.* [40] propose a method consisting of an aggregation of other approaches including single linkage, complete linkage, average linkage, K-means and Ward's clustering. The dataset proposed in [40] is known to be difficult and the clustering outcome is usually imperfect. It has $N = 788$ particles and

contains narrow bridges between clusters and uneven-sized clusters that are known to create difficulties for the clustering algorithms. The aggregation method gives seven clusters. The K-means method is unstable and gives very different clusterings for this example. Figure 2 shows two different outcomes of the MATLAB K-means program using as parameter, a number of clusters equal to 7.

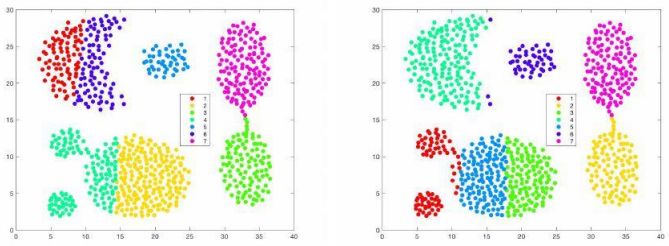

Figure 2: Example from [40] of $N = 788$ *particles: Two K-means clusterings.*

Our quantum block method (with $k = 9$, $\sigma = 3.6889$) gives also seven minima and thus seven clusters. Figure 3 Shows 6 drawing : The first drawing is the initial data. In the second one, the black dots corresponds to the new set of weighted particles obtained by using the block method with parameters $k = 9$ and $l = 1$ (Consequently, σ becomes $\sigma = 3.6889$). The red and green curves correspond to the implicit functions of M_{Bls} and L_{Bls} (The scale has been modified here following the variable changes proposed in Section 2). The determination of the clusters is done here from the minima using the Euclidean distance. Unfortunately, it faces some difficulties and some improvements could be done by using a spectral clustering. We use here a ϵ-neighborhood graph to produce the spectral clustering as shown in the second line of Figure 3. The MATLAB algorithm used needs as input the data *and* the number of clusters. First, we see the level lines and the clusters of the block data. The last drawing gives the rebuilding of the clustering on the initial data. It shows that the quality of the clustering is similar to the one of the aggregation of five different clustering approaches (see [40])

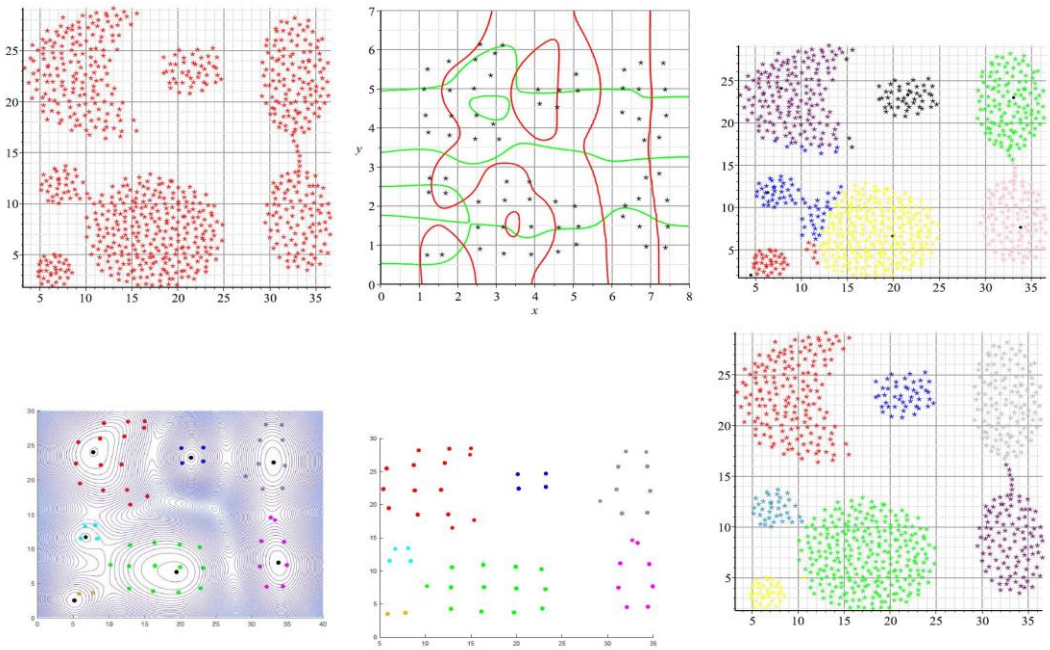

Figure 3: Example from of $N = 788$ particles. From left to right, the initial data, the main characteristics of the corresponding block method of parameters $k = 9$ and $l = 1$, the clustering using the Euclidean distance from the computed center (in black). Second line: The level line of the block potential energy function, new clustering based

on the spectral clustering method on the block data, reconstruction of the clustering on the initial data. The points in black in the contour plot show the cluster centers.

Fu and Medico [41] present an algorithm named FLAME dedicated to genome data clustering. The first step consists in the extraction of local structure information and identification of a "Cluster Supporting Object" (CSO). In this step, the proximity between each object and its k-nearest neighbors is used to calculate the object density. Here, this particular step is replaced by the computation of minima. We find two minima which correspond to the two CSOs presented in Ref. [41]. In Figure 4, the original data of $N = 240$ particles are presented to the left. The drawing in the middle shows the particles computed thanks to the block method and the associated implicit function with $k = 8$ and $l = 2$. Two minima are obtained: $(7.044805245, 25.30024272)$ and $(7.359015753, 17.11035066)$ Finally, the right plot gives the clustering. Note that in this specific example, the Euclidean distance is not the best choice to construct the clusters from the minima and a computation based on the spectral clustering is again needed to improve the result. This clustering is shown Figure 4.

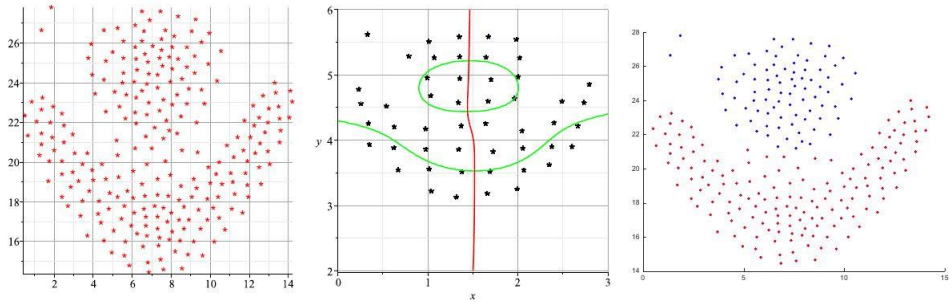

Figure 4: Example from [41] of 240 particles. From left to right: The initial data, the main characteristics of the block algorithm, the clustering based on the spectral clustering method

In our last example, we revisit the industrial optimization problem of Draper and Smith [3] [41] which originally used the maximin clustering method [43, p.1425]. The problem itself consisted of reported percentages of properly sealed bars of soap, sealer plate clearance x_1 and sealer plate temperature x_2. This is a fairly self-contained problem of two variables in 16 measured pairs:
$x_1 = (130,174,134,191,165,194,143,186,139,188,175,156,190,178,132,148)$,
$x_2 = (190,176,205,210,230,192,220,235,240,230,200,218,220,210,208,225)$.
Although the maximin method uses non-linear (constrained) optimization often requiring computational tools, such as linear programming, the conceptually simpler quantum clustering method nonetheless yields the same clusters. These are shown in Figure 5 using the same value of $\sigma = \frac{80}{4\sqrt{2}} = 14.1421$ ... as used in Ref. [3] to reveal four clusters. For the MATLAB spectral clustering, we used the 'seuclidean' option for the distance metric i.e. the standardized Euclidean distance (each coordinate difference between observations is scaled by dividing by the corresponding element of the standard deviation computed from the x_1, x_2 pairs). Also note that in three consecutive calls to MATLAB's Kmeans program yields three distinct clusterings for which only the first is correct, i.e. agrees with the quantum clustering, spectral clustering and maximin result.

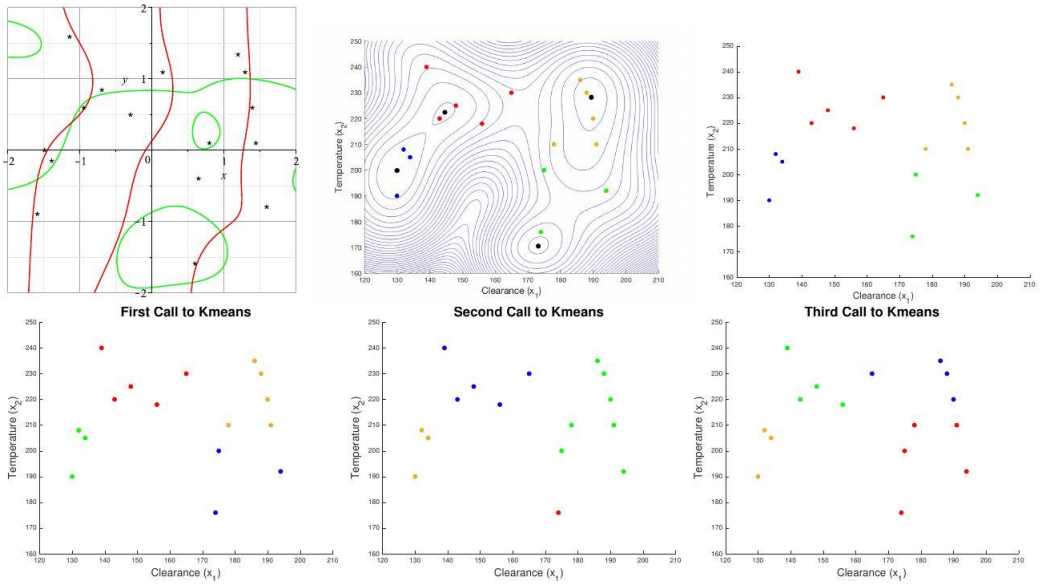

Figure 5: Quantum clustering (The data and the implicit functions of $M = 0$ and $L = 0$ after centralization and normalizalion with $\sigma = 14.1421$, the level lines and the clustering), MATLAB Spectral clustering using the standardized Euclidean distance and three consecutive calls to MATLAB's Kmeans clustering on the Smith and Draper pairs. As before, the points in black in the contour block show the cluster centers. Note that only the first Kmeans clustering agrees with both the quantum clustering and MATLAB's spectral clustering.

Unfortunately, some specific shapes such as ring-shaped or spiral-shaped clusters are challenging for numerous clustering methods including our QC block method. To overcome this issue, an approach based on optimization of an objective function, is proposed in [44] to detect specifically elliptic ring-shaped clusters. However, this approach is not appropriate when different kind of shapes coexist as for example in the case of Zahn's compounds [45]. It also requires a skilled operator to visualize the clusters. It will be a great challenge to improve the QC approach in order to detect such shapes.

4.6 Perspectives

In spite of claims to the contrary [46], even with extensions, K-means is no longer state-of-the-art. A means of finding *all* the potential minima of the quantum potential and consequently the number of clusters for a given range of σ is an essential key feature for data clustering under program control without prior visualization whilst K-means and even MATLAB's spectral clustering require the number of cluster centers on input and thus skilled operators. The quantum clustering approach yields this number for a given range. Automatic Data clustering under program control allows the processing of much bigger and more complex mixed datasets potentially providing a more robust industrial standard. It would multiply the number of platforms with large data collection tools such as Hadoop or MongoDB and thus a greater realization of patents for name of object disambiguation [1].

5. CONCLUSIONS

Herein, we have made considerable progress in dealing with the outstanding problem of getting all the centers of the quantum clustering method, namely finding *all* the minima of the quantum potential of Function (1) where σ is the standard deviation of the Gaussian distribution. The extrema of this potential are the roots of two coupled equations, which in principle are impossible to solve analytically. After simplifications, those equations become bivariate exponential polynomial equations and a number of useful properties have been proved. More precisely, limits of implicit function branches are given and the case of two particles is analytically solved. We also proved that the coupled equations have only one minimum if

the data are included in a square of side σ. This bound is directly useful to propose a new approach "per block". This technique decreases the number of particles by approximating some groups of particles to weighted particles. The minima of the corresponding coupled equations are then given numerically by which the number of clusters is obtained. Those minima can be used as cluster centers. However, for some complex examples, other clustering approaches such that spectral clustering gives better visual results (though they still require the number of clusters on input). On such examples, the approach consisting in the use of the block method (for the number of clusters but also for the weighted particles) gives very good results. Example 2, from Gionis *et al.* shows that the quality of the clustering is similar to the one of the aggregation of five approaches.

The approach used here is potentially useful for other types of exponential polynomials found in numerous Physical applications such as, for example, quantum mechanical diffusion Monte-Carlo calculations, where a precise knowledge of the nodal lines ensures accurate energy eigenvalues.

REFERENCES

[1] M. Fertik, T Scott and T Dignan, US Patent No. 2013/0086075 A1, Appl. No. 13/252,697 - Ref. US9020952B2, (2013)

[2] D. Horn and A. Gottlieb, Phys. Rev. Lett. **88**, 18702 (2002)

[3] T. C. Scott, M. Therani and X. M. Wang, Mathematics **5**, 1-17 (2017)

[4] A. Ben-Hur, D. Horn, H. T. Siegelmann and V. Vapnik, J. Mach. Learn. Res. **2**, 125-137 (2002)

[5] A. Messiah, Quantum Mechanics (Vol. I), English translation from French by G. M. Temmer, North Holland, John Wiley & Sons, Cf. chap. IV, section III. chap. 3, sec.12, 1966.

[6] A. Lüchow and T. C. Scott, J. Phys. B: At. Mol. Opt. Phys. **40**, 851-867 (2007)

[7] A. Lüchow, R. Petz R and T. C. Scott, J. Chem. Phys. **126**, 144110-144110 (2007)

[8] T. C. Scott, A. Lüchow, D. Bressanini and J.D. Morgan III, Phys. Rev. A (Rapid Communications) **75**, 060101 (2007)

[9] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, et al., Gaussian Inc., Wallingford CT (2009)

[10] T. C. Scott, I. P. Grant, M. B. Monagan and V. R. Saunders, Nucl. Instruments and Methods Phys. Research **389A**, 117-120 (1997)

[11] T. C. Scott, I. P. Grant, M. B. Monagan and V.R. Saunders, MapleTech **4**, 15-24 (1997)

[12] C. Gomez and T. C. Scott, Comput. Phys. Commun. **115**, 548-562 (1998)

[13] Achatz, M., McCallum, S., & Weispfenning, V. (2008). Deciding polynomial-exponential problems. In D. Jeffrey (Ed.), ISSAC'08: Proceedings of the 21st International Symposium on Symbolic and Algebraic Computation 2008 (pp. 215-222). New York: Association for Computing Machinery. https://doi.org/10.1145/1390768.1390799

[14] A. Maignan, Solving One and Two-dimensional Exponential Polynomial Systems, ISSAC98, ACM press, pp 215-221.

[15] Scott McCallum, Volker Weispfenning, Deciding polynomial-transcendental problems, Journal of Symbolic Computation, Volume 47, Issue 1, 2012, Pages 16-31, ISSN 0747-7171, https://doi.org/10.1016/j.jsc.2011.08.004.

[16] J. F. Rodriguez-Nieva and M. S. Scheurer, Identifying topological order through unsupervised machine learning, Nature Physics, Nature, Physics **15**, 790 (2019)

[17] Eran Lustig, Or Yair, Ronen Talmon, and Mordechai Segev, Identifying Topological Phase Transitions in Experiments Using Manifold Learning, Phys. Rev. Lett. **125**, 127401 (2020)

[18] Jielin Wang, Wanzhou Zhang, Tian Hua and Tzu-Chieh Wei, Unsupervised learning of topological phase transitions using Calinski-Harabaz score, accepted by Physical Review Research, (2020)

[19] Shervan Fekri Ershad, Texture Classification Approach Based on Energy Variation IJMT **2**, 52-55 (2012)

[20] Fan Decheng, Song Jon, Cholho Pang, Wang Dong, CholJin Won, Improved quantum clustering analysis based on the weighted distance and its application, Heliyon, Volume 4, Issue 11, 2018, e00984, ISSN 2405-8440, https://doi.org/10.1016/j.heliyon.2018.e00984.

[21] A. Maignan, On Symbolic-Numeric Solving of Sine-Polynomial Equations, Journal of Complexity, vol 16, Issue 1, (2000), pp. 274-285

[22] A. Maignan and T. C. Scott, SIGSAM **50**, 45-60 (2016)

[23] https://proofwiki.org/wiki/Upper_Bound_of_Natural_Logarithm
[24] B. Ripley, Cambridge University Press , Cambridge, UK (1996)
[25] B. Ripley, Available online , http://www.stats.ox.ac.uk/pub/PRNN/ (accessed on 3 January 2017)
[26] L. Bernardin, P. Chin, P. DeMarco, K. O. Geddes, D. E. G. Hare, K. M. Heal, G. Labahn, J. P. May, J. McCarron, M. B. Monagan, D. Ohashi and S. M. Vorkoetter, MapleSoft , Toronto (2012)
[27] I. Mezo and A. Baricz, On the generalization of the Lambert W function, Transactions of the American Mathematical Society 369, 7917-7934 (2017).
[28] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, Advances in Computational Mathematics **5**, 329-359 (1996)
[29] T. C. Scott, G. J. Fee and J. Grotendorst, SIGSAM **47**, 75-83 (2013)
[30] T. C. Scott, G. J. Fee, J. Grotendorst and W.Z. Zhang, SIGSAM **48**, 42-56 (2014)
[31] T. C. Scott and A. Maignan, SIGSAM **50**, 45-60 (2016)
[32] T. C. Scott, A. Dalgarno and J. D. Morgan III, Phys. Rev. Lett. **67**, 1419-1422 (1991)
[33] T. C. Scott, J. F. Babb, A. Dalgarno and J. D. Morgan III, J. Chem. Phys. **99**, 2841-2854 (1993)
[34] T. C. Scott, M. Aubert-Frécon and J. Grotendorst, Chem. Phys. **324**, 323-338 (2006)
[35] T. C. Scott and R. B. Mann and R. E. Martinez, AAECC (Applicable Algebra in Engineering, Communication and Computing) **17**, 41-47 (2006)
[36] P. S. Farrugia, R. B. Mann, and T. C. Scott. Class. Quantum Grav **24**, 4647-4659 (2007)
[37] Thanh-Toan Pham, PhD Thesis in Electronics, University of Grenoble Alpes (ComUE) , (2017)
[38] Exoplanet.eu-Extrasolar Planets Encyclopedia, Available online , http://exoplanet.eu/ Retrieved 16 November 2015 (accessed on 2 January 2017)
[39] A. Maignan, T. C. Scott, Quantum Clustering Analysis: Minima of the Potential Energy Function, *ISSN : 2231 - 5403* ,Vol. 10 – vol. NO: 19 - Issue: 19/12/2020.
[40] A. Gionis, H. Mannila, and P. Tsaparas, Clustering aggregation. ACM Transactions on Knowledge Discovery from Data (TKDD), 2007. 1(1): p. 1-30.
[41] L. Fu and E. Medico, FLAME, a novel fuzzy clustering method for the analysis.
[42] N.R. Draper and H. Smith, “Applied Regression Analysis”, 2nd ed., Wiley, New York, (1981).
[43] Forrest R. Miller, James W. Neill and Brian W. Sherfey, “Maximin Clusters from near-replicate Regression of Fit Tests”, Ann. Stat. **26**, no. 4, pp. 1411-1433, (1998).
[44] Isak Gath and Dan Hoory, Fuzzy clustering of elliptic ring-shaped clusters, Pattern Recognition Letters", Vol. 16, 1995, p. 727-741, https://doi.org/10.1016/0167-8655(95)00030-K.
[45] http://cs.joensuu.fi/sipu/datasets/
[46] A. Ahmad and S. S. Khan, "Survey of State-of-the-Art Mixed Data Clustering Algorithms," in IEEE Access, vol. 7, pp. 31883-31902, 2019, doi: 10.1109/ACCESS.2019.2903568.

Aude Maignan

Aude Maignan received the Ph.D. degree in applied mathematics from Limoges University, France, in 2000. She is an Associate Professor at Université Grenoble Alpes, France. Her research interests include complex systems, generalized Lambert function and graph rewriting.

Tony Scott

Tony C. Scott graduated in 1991 with a Ph.D. in Theoretical Physics at the University of Waterloo (1991). As an N.S.E.R.C. postdoctoral fellow, he worked at the Harvard-Smithsonian Center for Astrophysics in Cambridge MA USA (90-92) and the Mathematical Institute at Oxford University in the UK ('93-'95). This was followed by further academic work in other countries until he became a Data Scientist for 7 years in Silicon Valley USA. After being a Physics professor in China, he is back in the private sector.