DACCO: A Discrete Ant Colony Algorithm to Cluster Geometry Optimization

Nuno Lourenço
CISUC, Department of Informatics Engineering
University of Coimbra, 3030 Coimbra, Portugal
naml@dei.uc.pt

Francisco B. Pereira
[1]CISUC, Department of Informatics Engineering
University of Coimbra, 3030 Coimbra, Portugal
[2]ISEC, Quinta da Nora, 3030 Coimbra, Portugal
xico@dei.uc.pt

ABSTRACT

We present a discrete ant colony algorithm to cluster geometry optimization. To deal with this continuous problem, the optimization framework includes functions to map solutions across the discrete and continuous spaces. Results obtained with short-ranged Morse clusters show that the proposed approach is effective, scalable and is competitive with state-of the-art optimization methods specifically designed to tackle continuous domains. A detailed analysis is presented to help to gain insight into the role played by several components of the ant colony algorithm.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Global Optimization*; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Heuristic Methods*

General Terms

Algorithms

Keywords

Ant colony optimization, Cluster geometry optimization, Hybridization, Morse clusters

1. INTRODUCTION

Ant Colony Optimization (ACO) methods are population-based metaheuristics regularly applied in combinatorial and continuous optimization problems [7]. The original algorithm, Ant System (AS), was proposed by Dorigo in the early 1990's [5] and it is loosely inspired by the organization of natural ant societies. When foraging, ants deposit pheromone on the ground to guide co-workers towards promising areas. Then, individuals belonging to the colony communicate via stigmergy, an indirect way to exchange information mediated by modifications in the environment.

GECCO'12, July 7-11, 2012, Philadelphia, Pennsylvania, USA.

Following AS, many other ACO variants have been proposed in the literature. Relevant examples are the Ant Colony System (ACS) [6] and the Max-Min Ant System (MMAS) [19], two approaches that have been thoroughly tested on combinatorial optimization problems. Even though earlier ACO research focused on discrete problems, there are a number of variants for continuous optimization. Bilchev and Parmee pioneered the extension of an ant algorithm to explore real-valued search spaces [1] and, since then, several other approaches have been proposed [13, 18, 21].

In this paper we propose an ACO algorithm to solve the cluster geometry optimization problem. This is an important problem from the chemistry area, where the goal is to find the optimal structural organization for a set of particles in a 3D space [8, 23]. Estimating the most relevant properties of chemical clusters has immediate relevance in many areas, ranging from protein structure prediction to the study of the influence of stratospheric clouds in ozone destruction. Also, a proper understanding of cluster properties is crucial for the field of nanotechnology. In simple terms, a cluster is an aggregate of between a few and millions of atoms or molecules, which may present distinct physical properties from those of a single particle or bulk matter. The Potential Energy Function (PES) models the interactions between particles and it contains all the relevant information about the chemical system. PES are multidimensional functions generating highly roughed search landscapes, with a number of local minima increasing exponentially with the size of the cluster [22]. The goal of the optimization is to find the global minimum of the PES, i.e., to determine the relative position in the 3D space of all the atoms that correspond to the lowest potential energy. In our study we adopt the Morse function to describe interactions between atoms, a model potential that can accurately represent real materials [3, 9, 16]. Moreover, this is a difficult real-valued optimization function that has been regularly used as a benchmark to determine the performance of global search methods for cluster geometry optimization [4, 10, 11, 17].

We have developed DACCO (Discrete Ant Colony algorithm to Cluster geometry Optimization), a framework that follows an unusual approach to apply an ACO algorithm to seek for the optimal arrangement of atomic clusters. We discretize the problem and rely on the MMAS variant to find solutions on the discrete space. Then, a mapping function, consisting of a gradient-driven local optimization procedure, pushes the solutions obtained by the ants back to the original continuous space. The reasons for relying on this framework are twofold: first, there is a deeper insight into the

main properties of existing discrete ACO algorithms, when compared to continuous variants; on the other hand, the problem consists in placing objects in a 3D space, subject to a minimal distance constraint. Then, discretizing the space is a straightforward strategy to ensure a valid arrangement of the cluster. A previous work from Korb et al. adopts a similar strategy to solve the protein-ligand docking problem [12].

We test the effectiveness of DACCO by seeking the global optima of short-ranged Morse clusters between 30 and 80 atoms. Results obtained show that the hybrid architecture composed by the discrete MMAS variant and the continuous local optimization procedure is competitive with current state-of-the-art metaheuristics for cluster geometry optimization. Additionally, we investigate how some specific components of DACCO help to enhance the likelihood of discovering good quality solutions.

The paper is structured as follows: in section 2 we provide a general description of ACO algorithms, including the main variants for continuous optimization. Section 3 comprises a presentation of DACCO and of the optimization problem to be addressed. Optimization results and the corresponding analysis are accomplished in section 4 and, finally, section 5 gathers the main conclusions and suggests directions for future work.

2. ANT COLONY OPTIMIZATION

The first ACO variants (e.g., AS or ACS) were proposed for combinatorial optimization, such as the traveling salesperson or the quadratic assignment problems [5, 6, 7]. The application of an ant algorithm to a specific problem requires the specification of a set of solution components. A graph is then created, where each vertex represents a component and edges link related vertices. The artificial ants construct solutions by traversing this graph. They start at a random vertex and iteratively collect new components. At each vertex of the path they build, ants stochastically select a new edge. The probability of choosing a specific edge (i.e., the desirability of adding a particular new component to the solution) depends on static heuristic information and on the pheromone level of that connection. Higher pheromone concentrations signal components that tend to appear in better solutions. When all artificial ants have completed their solutions, the pheromone levels are updated. To avoid stagnation, evaporation first decreases all pheromone values by a given factor. Then, a subset of ants deposits pheromone on the edges they crossed when building a solution. The amount placed by each ant is proportional to the quality of the solution found. ACO variants differ in the way ants build solutions and/or on the pheromone update step. For algorithmic details, consult the aforementioned references. By iterating these steps until a termination criterion is met, a solution to the problem will emerge from the cooperation made by the ants. In algorithm 1 we present the general structure of an ACO metaheuristic. The *Daemon actions* step identifies several optional actions, such as the application of local search or pheromone matrix restart.

Algorithm 1 General ACO Algorithm

```
Set parameters
Init pheromone trails
while termination condition not met do
  Construct solutions
  Update pheromones
  Daemon actions {optional}
end while
```

2.1 ACO for Continuous Optimization

There are several reports in the literature describing extensions of ACO architectures to continuous optimization. In 1995, Bilchev and Parmee [1] proposed an algorithm where the ants start from a given location in the search space (the nest) and explore the continuous neighborhood by selecting one from a set of vectors. These vectors depart from the nest and represent a set of possible search directions. The probability of selecting a vector reflects the likelihood that this direction leads to promising solutions. Tsutsui presented an approach based on the concept of aggregate pheromone density functions [21]. Each ant emits an aggregate pheromone amount, which is proportional to quality of the solution found. The total aggregate pheromone density is therefore biased towards promising areas of the search space and is used by a probability density function to generate new solutions. $ACO_{\mathbb{R}}$ is a recent approach from Socha and Dorigo to tackle continuous problems [18]. It relies on a set of solutions (the solution archive) to represents the pheromone model. The archive is used to bias the generation of new solutions. In this process, each variable is treated independently and its value is obtained by sampling a mixture of weighted Gaussian functions. Recently, this method was enhanced with the addition of a variable size solution archive and a local search procedure [13]. PLANTS is an ACO algorithm for protein-ligand docking proposed by Korb et al [12]. In this problem, the goal is to find the optimal settings for several real valued parameters defining the degrees of freedom and torsional degrees of freedom for the ligand and protein, respectively. The approach followed in PLANTS is similar to the one adopted in our research. The continuous variables are discretized and the MMAS variant is applied to solve the problem. PLANTS does not use heuristic information and relies on a local search algorithm to map solutions back to the continuous space.

3. DACCO

In this section we present DACCO, the Discrete Ant Colony algorithm that will be applied to Cluster geometry Optimization. First, we describe the problem to be addressed and then detail the main algorithmic components of the optimization framework.

3.1 Morse Clusters

A chemical cluster is an aggregate of between a few and millions of atoms or molecules. Understanding the properties of clusters is relevant for many areas, from protein structure prediction to the field of nanotechnology. The PES is a multidimensional function that contains all the relevant information about the chemical system [23]. The goal of cluster geometry optimization is to determine the optimal structural organization for the particles that compose a given aggregate, i.e., to discover the arrangement in the 3D space that corresponds to the global minimum of the PES. The Morse function [16] is a model potential regularly adopted in cluster geometry optimization, as it provides accurate approximations of real materials (e.g., alkali-metal clusters or C_{60} molecules) [3]. It is a pairwise additive

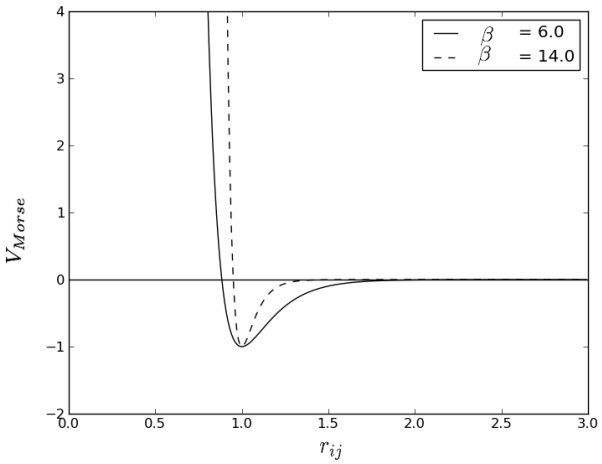

Figure 1: Morse Potential for different values of β

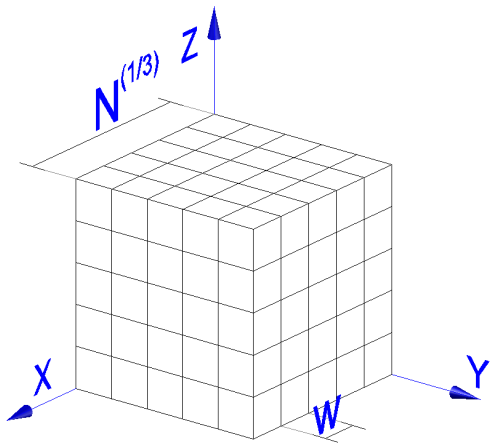

Figure 2: Discretized Cluster Geometry Optimization Space

potential that considers the distance between every pair of particles to determine the energy of the cluster. In Morse clusters, all particles are identical. Then, the total potential energy of a N-cluster Morse aggregate is given by the following equation:

$$V_{Morse} = \epsilon \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left(\exp^{-2\beta(r_{ij}-r_0)} - 2\exp^{-\beta(r_{ij}-r_0)} \right) \quad (1)$$

where r_{ij} is the Cartesian distance between atoms i and j, ϵ is the bond dissociation energy, r_0 is the equilibrium bond and β is the range exponent of the potential. Following [9], both ϵ and r_0 are set to 1.0, leading to a scaled version of the Morse function without specific atom interactions.

Hence, the potential has a single adjustable parameter β that determines the shape of the energy contribution of every pair of atoms [23]. The chart from figure 1 illustrates how this pairwise contribution is modeled as a function of the distance between atoms (displayed in the xx axis). Two different β values are exemplified: in both cases, the lowest potential energy is achieved when the two atoms are separated by a distance that is equal to the equilibrium bond. However,moving from a long ranged potential ($\beta = 6.0$) to a short ranged version ($\beta = 14.0$) leads to a narrower curvature, promoting the appearance of very roughed landscapes, with the number of local minima increasing exponentially as the number of atoms increases [8, 22]. In our study β is set to 14.0, as this creates optimization instances that are particularly challenging.

3.2 Optimization Framework

When solving a Morse cluster instance with N identical atoms, one must specify the location of each particle. The search space is defined as a cubic area in the first quadrant with side $N^{1/3}$, as this enables the cluster volume to scale correctly with N [11]. Additionally, small interatomic distances should not be allowed, because the potential tends to infinity when two atoms are too close (see figure 1). This minimum distance constraint creates a favorable environment for the application of discrete ACO variants. Ants build a solution by traversing the 3D space and placing atoms in selected locations. A careful definition of the discrete step length ensures that two atoms never stay too close to each other.

In concrete, the problem of cluster geometry optimization is defined in a 3 dimensional space $(x, y, z) \in \mathbb{R}^3$, where $x, y, z \in [0, N^{1/3}]$. This cube is divided into cells of side w. The value selected for this parameter must ensure that it is possible to place atoms in neighbor locations, but it is not feasible to have two particles inside the same cell. Given the value of parameter r_0, for all experiments reported in this work, w is set to 0.6. We did some additional tests with different values and verified that this setting provides the best compromise. Figure 2 presents an overview of the discretized optimization space.

The well-known MMAS discrete variant [19] is adopted by DACCO to seek for the optimal configuration of different Morse cluster instances. The automatic rescaling approach proposed by the Hyper-Cube Framework (HCF) [2] is incorporated in the algorithm, in order to increase its robustness. Algorithm 2 presents a general overview of DACCO, with an identification of the steps required to move from the continuous space to the discrete one and vice-versa. In the next paragraphs we detail the operations involved in the application of this algorithm.

Algorithm 2 General Overview of DACCO

```
Construct Discrete Search Space
Initialize Pheromone Matrix
while termination condition not met do
    Construct Solutions in the Discrete Space
    Convert Solutions to the Continuous Space
    Evaluate Solutions in the Continuous Space
    Convert Solutions to the Discrete Space
    Apply Discrete Local Improvement
    Update Pheromone Values
end while
return best individual in the population
```

Initialize pheromones: In cluster geometry optimization, what is important is the relative position of the atoms in space, and not the order in which they are placed. Then,

the structure of the pheromone matrix is identical to that of the discretized cube and it stores the desirability of placing an atom in a given cell. In the beginning of the optimization, all the cells from the pheromone matrix are initialized with the same value. Following the recommendations from [2], we set this value to 0.5.

Algorithm 3 Solution Construction

```
Given an ant do:
current_location = select_random_cell()
place_atom()
placed_atoms = 1
while placed_atoms < N do
  neighbors = find_feasible_neighborhood(ant)
  current_location = find_next_cell(neighbors, pher_matrix)
  place_atom()
  placed_atoms = placed_atoms + 1
end while
```

Construction of Solutions: Algorithm 3 details the actions performed by an ant to build a solution. When seeking for a good geometry of a Morse instance with N atoms, an ant starts in a randomly selected cell and must visit another N-1 locations to place one atom in each one of them. Ants consider both heuristic information and pheromone values to select the next cell to visit. The first component favors cells in the neighborhood of already occupied locations to place the next atom. Optimal configurations of clusters tend to correspond to compact structures, so heuristic information bias the formation of solutions where atoms are located in adjacent cells (the discretization of the space prevents atoms from being too close). We consider two alternatives for the definition of the heuristic neighborhood:

Current Moore Neighborhood It consists of all unoccupied cells that surround the current location of the ant. Different neighborhood ranges R can be considered, impacting the size of the cube around the current cell. In panel a) of figure 3 we illustrate a Current Moore Neighborhood with R=2. For simplicity, the example is presented in a 2D grid.

Full Moore Neighborhood It consists of all unoccupied cells surrounding atoms already placed in the grid. Just like in the previous alternative, different neighborhood ranges may be considered. Panel b) of figure 3 displays a Full Moore Neighborhood with R=1.

After determining the feasible neighborhood, an ant selects the next cell to place an atom by applying the MMAS selection rule. In this stage, only the pheromone values are considered (MMAS selection does not take into account any heuristic information). This process is repeated until all the atoms are placed in the grid.

Mapping Solutions to the Continuous Space and Evaluation: Mapping of a discrete solution into the continuous space consists in two steps: first, each atom is positioned in the geometric center of the cell in which it was placed during the construction of the solution. Afterwards, the positioning of atoms inside the cluster is refined by the application of a local optimization procedure. Most effective unbiased techniques for cluster geometry optimization are hybrid approaches combining a global optimization algorithm with a local search procedure [10, 11, 17]. In accordance, we

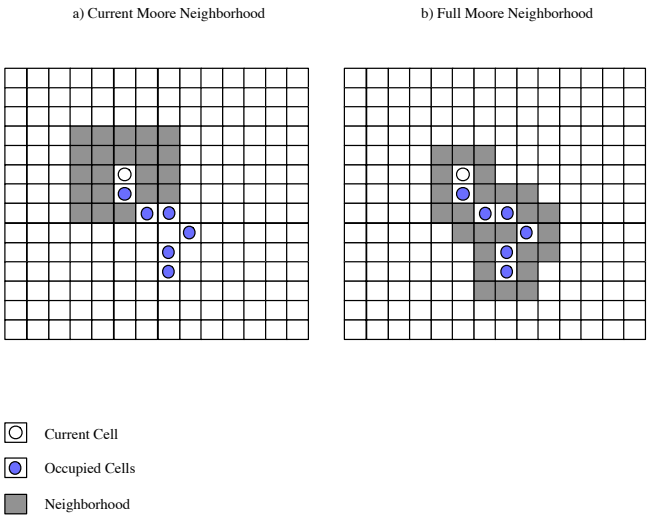

Figure 3: Examples of heuristic neighborhoods in a 2D grid: a) Current Moore neighborhood with R=2; b) Full Moore Neighborhood with R=1.

rely on the Broyden-Fletcher-Goldfarb-Shannon (L-BFGS) method [14], a conjugate gradient algorithm that efficiently guides the solution created by the ants into the nearest local minimum. Finally, the resulting solution is evaluated with equation 1.

Mapping Solutions to the Discrete Space After evaluation, clusters are mapped back to the discrete space. To accomplish this, atoms are simply moved to the geometric center of the cell where they were positioned by L-BFGS. Given the selected w value, it is highly unlikely that local optimization places two atoms in the same cell. Anyway, if this happens, then one of the particles is moved to the nearest unoccupied cell. There are two reasons to discretize the solutions after evaluation. The first is to apply a discrete local improvement method, that seeks for better configurations by iteratively perturbing existing solutions. The improvement strategy is straightforward: it seeks for the worst atom of the cluster (i.e., the one that has the highest contribution to the total potential energy) and moves it to an unoccupied cell randomly selected. The modified solution is locally optimized with L-BFGS and evaluated. If the resulting cluster has lower energy than the original one, then it is kept. The process is repeated for a predetermined number of iterations. The second reason to move the solutions back to the discrete space is to perform the update of the pheromone matrix. First, evaporation slightly decreases all values. Then, the rule described in [2] for MMAS is used to reinforce the pheromone of promising cells. Pheromone reinforcement adopts a diffusion model. If a cell is going to be reinforced with a given amount A, then it only receives $(A - p)$ and the quantity that is left over is distributed in equal parts among all adjacent cells. This procedure aims to smooth the transition between the discrete and continuous spaces and it was also adopted by [12].

4. EXPERIMENTS

In this section we present the results obtained by DACCO

in the optimization of short-ranged Morse clusters between 30 and 80 atoms. With these experiments we aim to determine the absolute optimization performance of the proposed approach and to gain insight into the role played by the different algorithmic components. The settings of the algorithm are the following: Number of runs: 30; Number of ants: equal to the number of atoms of the instance; Evaluations: 5,000,000; α: 4; Pheromone diffusion p: 0.5; Current Moore neighborhood with R=3; Discrete local improvement iterations: 10; L-BFGS accuracy: $1.0E-8$. Each iteration performed by the L-BFGS method counts as one evaluation.

Analysis of results focus on the ability of DACCO to find the putative global optimum (within a pre-specified computational effort), as this is the criterion regularly adopted in cluster geometry optimization studies. Therefore, for each instance we present the success rate (SR) of DACCO, i.e., the number of runs (out of 30) that discovered the best known solution. For completeness, we also provide the mean best fitness (MBF) of all reported experiments.

4.1 Optimization Results

Table 1 contains an overview of the optimization results. There is a line for each one of the instances selected. Column *Opt* displays the potential energy of the putative global optimum[1], whereas columns 3 and 4 present the SR and the MBF, respectively. The last column *Deviation* measures how the MBF deviates, in percentage, from the best-known solution. Results confirm that DACCO can effectively find good quality solutions. With just two exceptions, $N = \{47, 79\}$, it always discovers the putative global optima of short ranged Morse instances between 30 and 80 atoms. The effectiveness of DACCO is particularly relevant, since the global optima of different Morse instances correspond to distinct structural shapes (e.g. poly-tetrahedral, icosahedral or decahedral) [9]. The outcomes displayed in table 1 suggest that DACCO is not biased towards generating solutions with a particular geometry.

There is a trend for a decrease in the success rate achieved by DACCO as the clusters grow in size. This effect can be partially justified by the fixed number of evaluations granted to all experiments reported in this paper. Instances with more atoms define larger search spaces and, therefore, it is expected that the success rate of DACCO might be lower. Additionally, the optimal geometry for some specific Morse instances is particularly hard to discover, which helps to explain the variation in the success rate. Examples of magic numbers (*i.e.*, particularly hard Morse instances) are 38, 47, 61, 68 or 79 atoms [9, 10].

The MBF achieved by DACCO in all instances is always close to the value of the putative optima. The deviation rarely exceeeds 1% and, in most cases, is well below this threshold. This result shows that the proposed algorithm is robust, since, even when it does not discover the global optimum, it converges to good quality local optima.

4.2 Comparison with Other Approaches

It is essential to compare the results obtained by DACCO with those achieved by state-of-the-art methods for cluster geometry optimization. We selected two global optimization algorithms recently described in the literature: a hybrid

[1]Consult the Cambridge Cluster Database for an up to date list of Putative Optima: http://www-wales.ch.cam.ac.uk/CCD.html

N	Opt	SR	MBF	Deviation
30	-106.835790	28 / 30	-106.831095	0.004
31	-111.760670	30 / 30	-111.760670	0.000
32	-115.767561	30 / 30	-115.767561	0.000
33	-120.741345	30 / 30	-120.741345	0.000
34	-124.748271	30 / 30	-124.748271	0.000
35	-129.737360	30 / 30	-129.737360	0.000
36	-133.744666	30 / 30	-133.744666	0.000
37	-138.708582	28 / 30	-138.682731	0.019
38	-144.321054	25 / 30	-144.053535	0.185
39	-148.327400	26 / 30	-148.243303	0.057
40	-152.333745	25 / 30	-152.228797	0.069
41	-156.633479	11 / 30	-156.483040	0.096
42	-160.641020	5 / 30	-160.449243	0.119
43	-165.634973	6 / 30	-165.361457	0.165
44	-169.642441	3 / 30	-169.383463	0.153
45	-174.511632	3 / 30	-174.295931	0.124
46	-178.519320	2 / 30	-178.371855	0.083
47	-183.508227	0 / 30	-183.095976	0.225
48	-188.888965	19 / 30	-188.402414	0.258
49	-192.898412	15 / 30	-192.675230	0.116
50	-198.455632	12 / 30	-197.853115	0.304
51	-202.468274	13 / 30	-201.992636	0.235
52	-207.480764	18 / 30	-207.294088	0.090
53	-211.493405	14 / 30	-211.225486	0.127
54	-216.636864	5 / 30	-216.171684	0.215
55	-220.646208	10 / 30	-220.422433	0.101
56	-225.655136	11 / 30	-224.935830	0.319
57	-230.663986	15 / 30	-229.976103	0.298
58	-234.809078	9 / 30	-234.394252	0.177
59	-240.572493	6 / 30	-239.338941	0.513
60	-244.579066	5 / 30	-244.090244	0.200
61	-249.587740	3 / 30	-248.668557	0.368
62	-253.612942	11 / 30	-253.277141	0.132
63	-258.620607	7 / 30	-258.032594	0.227
64	-264.587042	6 / 30	-262.706618	0.711
65	-268.594702	11 / 30	-267.406606	0.442
66	-273.602343	6 / 30	-272.023822	0.577
67	-278.400953	2 / 30	-276.976700	0.512
68	-282.683003	2 / 30	-281.715729	0.342
69	-287.462110	8 / 30	-286.366826	0.381
70	-292.462856	10 / 30	-291.443945	0.348
71	-298.405353	3 / 30	-295.263173	1.053
72	-302.413229	5 / 30	-299.983942	0.803
73	-307.421094	5 / 30	-304.684859	0.890
74	-312.441302	4 / 30	-309.237024	1.026
75	-318.407330	6 / 30	-315.446093	0.930
76	-322.414257	4 / 30	-319.405598	0.933
77	-327.371999	4 / 30	-324.342238	0.925
78	-331.379143	2 / 30	-328.915871	0.743
79	-336.798725	0 / 30	-333.181307	1.074
80	-340.811371	2 / 30	-338.162647	0.777

Table 1: Results obtained by DACCO in the optimization of Morse clusters between 30 and 80 atoms.

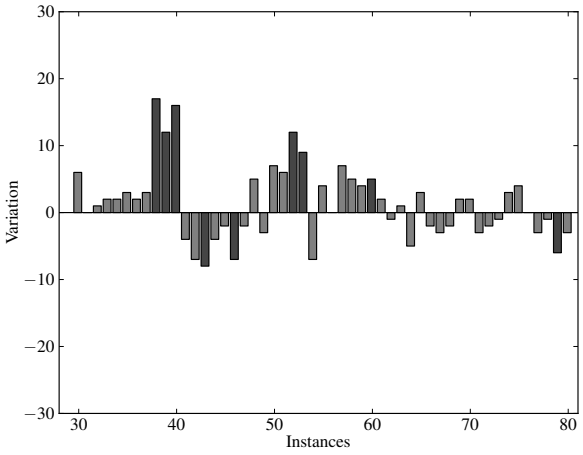

Figure 4: Comparison of the SR obtained by DACCO and the hybrid EA from [17] in the optimization of Morse clusters between 30 and 80 atoms. Darker columns highlight instances where significant differences were found.

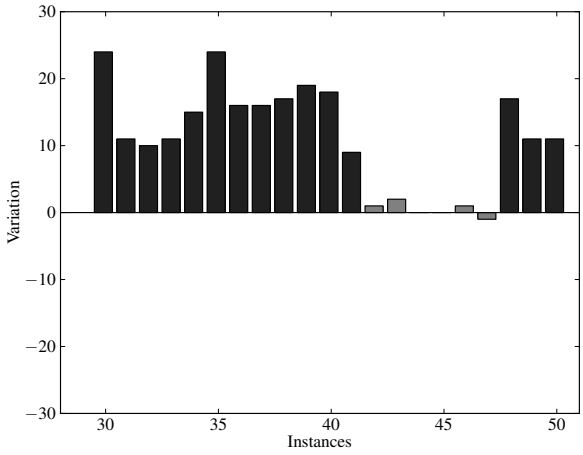

Figure 5: Comparison of the SR obtained by DACCO and the PSO from [15] in the optimization of Morse clusters between 30 and 50 atoms. Darker columns highlight instances where significant differences were found.

evolutionary algorithm (EA) [17] and a Particle Swarm Optimization (PSO) method [15]. The hybrid EA was the first completely unbiased algorithm to discover all putative global optima for short-ranged Morse clusters until 80 atoms. As for the PSO, it is the most effective swarm intelligence algorithm for this problem. Both algorithms incorporate a local procedure based on the L-BFGS method, similar to the one described for DACCO, and were granted the same number of evaluations to optimize the Morse instances selected for the comparison. For algorithmic details concerning both methods, consult the aforementioned references.

In figure 4 we display the SR difference between DACCO and the hybrid EA, across all instances. A positive (negative) variation corresponds to an instance where DACCO achieved a higher (lower) SR than the EA. In the same manner, figure 5 presents the comparison between DACCO and the PSO. Here, we only display results between 30 and 50 atoms, since the PSO was not applied to larger instances. A general overview of figure 4 reveals that the performance of DACCO and the EA is comparable. DACCO achieved a higher success rate on 27 instances, whereas EA was better on 21 instances (on the 3 remaining cases, both algorithms obtained the same SR). To strengthen the analysis, we applied a pairwise test to compare proportions, described in [20], to the results obtained by these two methods. Darker columns in the figure highlight instances where significant differences were found (significance level of 1%). The statistical analysis confirms the global equivalence of the two algorithms, as only in 9 instances a significant difference is identified (in 6 of them DACCO is better, whereas the EA achieves a higher SR in the remaining 3).

The chart from figure 5 clearly confirms that DACCO is more effective than the PSO when seeking for low energy Morse clusters. In 18 out of 21 instances the success rate achieved by DACCO is higher and, in most of them (15 instances), the difference if statistically significant. This is a relevant result, as it reveals that an adapted discrete ACO algorithm can outperform a PSO method in a hard continuous optimization problem.

As a final note, it is important to refer that both the EA and the PSO require the explicit application of a structural distance measure to help maintain the diversity in the population of solutions. Without this mechanism, both algorithms are likely to suffer from premature convergence. Results presented in this section show that DACCO is able to obtain results comparable to the hybrid EA without relying on an explicit and problem specific way to maintain diversity. Also, it does not require the computational overhead needed to calculate this measure.

4.3 DACCO Detailed Analysis

This last section aims to establish the impact of specific DACCO components on the performance of the algorithm. We focus our analysis on the definition of the heuristic neighborhood, on the pheromone diffusion model and on the application of the discrete local improvement method. Due to space constraints we present results obtained with a subset of instances $N = \{30, 50, 70, 79\}$, which help to gain insight into the relevance of specific design options. We did, however, additional tests with clusters with different sizes and verified that the results follow the same trend. Unless explicitly stated, experiments described in this section adopt the settings previously defined.

The heuristic is a greedy component that can play an important role in the behavior of an ACO algorithm. In what concerns DACCO, it bias the placement of atoms in the vicinity of already occupied positions. Recalling section 3, there are several options to define the neighbor cells that can be considered as viable options to place the next particle. Table 2 displays the optimization results obtained by several configurations in the selected instances. In concrete, we display the SR and the MBF of experiments done with the Current and Full Moore neighborhood and with $R = \{1, 2, 3, 4\}$. An overview of table 2 reveals that there

	Moore								Full Moore							
	1		2		3		4		1		2		3		4	
	SR	**MBF**	**SR**	**MBF**	**SR**	**MBF**	**SR**	**MBF**	**SR**	**MBF**	**SR**	**MBF**	**SR**	**MBF**	**SR**	**MBF**
30	25 / 30	-106.824	26 / 30	-106.826	28 / 30	-106.831	27 / 30	-106.828	25 / 30	-106.824	27 / 30	-106.828	27 / 30	-106.828	25 / 30	-106.824
50	0 / 30	-196.301	4 / 30	-196.958	12 / 30	-197.853	7 / 30	-197.898	2 / 30	-196.923	8 / 30	-197.620	7 / 30	-197.608	10 / 30	-197.819
70	0 / 30	-287.568	2 / 30	-289.673	10 / 30	-291.443	9 / 30	-291.483	0 / 30	-288.645	2 / 30	-290.621	8 / 30	-291.543	6 / 30	-291.487
79	0 / 30	-327.855	1 / 30	-332.482	0 / 30	-333.181	0 / 30	-333.618	0 / 30	-329.417	0 / 30	-332.539	1 / 30	-333.754	2 / 30	-334.085

Table 2: Optimization results obtained by different DACCO heuristic neighborhoods in selected Morse instances.

	No Pheromone Diffusion		With Pheromone Diffusion	
	SR	MBF	SR	MBF
30	26 / 30	-106.826	28 / 30	-106.831
50	7 / 30	-197.709	12 / 30	-197.853
70	10 / 30	-291.563	10 / 30	-291.443
79	0 / 30	-333.372	0 / 30	-333.181

Table 3: Impact of the pheromone diffusion model in the optimization results obtained by DACCO in selected Morse instances.

	No Local Improvement		With Local Improvement	
	SR	MBF	SR	MBF
30	2 / 30	-106.378	28 / 30	-106.831
50	0 / 30	-194.405	12 / 30	-197.853
70	0 / 30	-281.598	10 / 30	-291.443
79	0 / 30	-324.128	0 / 30	-333.181

Table 4: Impact of the discrete local improvement step in the optimization results obtained by DACCO in selected Morse instances.

are no noteworthy differences in the outcomes of the different heuristic neighborhoods (Current vs. Full). As for the range value, experiments done with $R = 3$ and $R = 4$ tend to obtain better results, suggesting that it is advantageous to define a neighborhood that is not too narrow. Overall, these tests show that DACCO is robust to small variations in the definition of the heuristic neighborhood.

Table 3 displays the optimization results obtained in the selected instances by two DACCO configurations: the standard algorithm with pheromone diffusion and a variant without this mechanism. An inspection immediately reveals that both variants obtained comparable results, suggesting that, with the framework proposed in this paper, it is not mandatory to adopt a diffusion model to smooth the transition between the discrete and continuous spaces. However, it is important to emphasize that this result was obtained with a specific experimental condition: Current Moore neighborhood with $R = 3$ and cell size $w = 0.6$. In future research we will verify if the similarity between results is maintained for other settings.

Table 4 shows the results obtained by DACCO with and without the discrete local improvement step. The number of evaluations is kept fixed in both configurations, so when DACCO is running alone it performs a higher number of iterations. A brief perusal of the table reveals the relevance of discrete local improvement. The version of DACCO that does not incorporate this component fails to discover the putative global optimum in nearly all runs of every selected instance. Also, the MBF values are clearly worse than those achieved by the full DACCO framework. These results show that DACCO clearly benefits from the addition of a discrete greedy perturbation of the solutions built by the ants. This performance improvement is in agreement with existing literature, as there are many reports describing the advantage of incorporating a local improvement step in the processing of an ACO algorithm [7].

5. CONCLUSIONS

We presented DACCO, a discrete ant colony algorithm for cluster geometry optimization. In this framework, the continuous 3D space where the cluster must be located is discretized and ants build a solution by iteratively traversing a 3D grid and placing atoms at selected cells. DACCO includes functions to map clusters between the discrete and continuous spaces and incorporates a discrete greedy local improvement step to further enhance solutions created by the ACO algorithm.

Results obtained with short-ranged Morse instances show that DACCO is competitive with state-of-the-art methods for cluster geometry optimization. It clearly outperforms a PSO approach recently proposed and its performance is comparable to the most effective hybrid EA for this problem. This outcome confirms that the application of discrete ACO variants to (discretized) hard continuous problems is a viable option and it allows the development of successful optimization methods.

There are several possible directions to follow in what concerns future work. On the one hand, our research efforts will be directed towards modifications that can further enhance the robustness and scalability of DACCO framework. This might allow its application to molecular aggregates (e.g, H_2O clusters), a particularly hard class of cluster geometry optimization problems. Also, we aim to investigate the generalization ability of DACCO, by applying it to other continuous optimization problems.

6. ACKNOWLEDGMENTS

This work was supported by Fundação para a Ciência e Tecnologia (FCT), Portugal, under project PTDC/QUI/69422/2006, which is financed by Programa Operacional Factores de Competitividade (COMPETE) of QREN and FEDER programs. The authors are grateful to the Laboratory for Advanced Computing of the University of Coimbra for the provision of supercomputer time on Milipeia (Project daion_10).

7. REFERENCES

[1] G. Bilchev and I. Parmee. The ant colony metaphor for searching continuous design spaces. In T. Fogarty, editor, *Evolutionary Computing*, Lecture Notes in Computer Science 993, pages 25–39. Springer, 1995.

[2] C. Blum and M. Dorigo. The hyper-cube framework for ant colony optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 34(2):1161–1172, 2004.

[3] P. Braier, R. Berry, and W. D. How the range of pair interactions govern features of multidimensional potentials. *Journal of Chemical Physics*, 93:8745–8756, 1990.

[4] L. Cheng and J. Yang. Global minimum structures of morse clusters as a function of the range of the potential: $81 \leq n \leq 160$. *Journal of Physical Chemistry A*, 111:5287–5293, 2007.

[5] M. Dorigo. *Optimization, Learning and Natural Algorithms*. PhD thesis, Politecnico di Milano, Italy, 1992.

[6] M. Dorigo and L. Gambardella. Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1):53–66, 1997.

[7] M. Dorigo and T. Stützle. *Ant Colony Optimization*. Bradford Company, 2004.

[8] J. P. K. Doye, R. H. Leary, M. Locatelli, and F. Schoen. Global optimization of morse clusters by potential energy transformations. *Informs Journal on Computing*, 16(4):371–379, 2004.

[9] J. P. K. Doye and D. J. Wales. Structural consequences of the range of the interatomic potential: a menagerie of clusters. *Journal of Chemical Society, Faraday Transactions*, 93:4233–4243, 1997.

[10] A. Grosso, M. Locatelli, and F. Schoen. A population-based approach for hard global optimization problems based on dissimilarity measures. *Mathematical Programming*, 110:373–404, 2007.

[11] R. L. Johnston. Evolving better nanoparticles: Genetic algorithms for optimising cluster geometries. *Dalton Transactions*, pages 4193–4207, 2003.

[12] O. Korb, T. Stützle, and T. Exner. An ant colony optimization approach to flexible protein-ligand docking. *Swarm Intelligence*, 1:115–134, 2007.

[13] T. Liao, M. Oca, D. Aydin, T. Stützle, and M. Dorigo. An incremental ant colony algorithm with local search for continuous optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2011)*, pages 125–132. ACM, 2011.

[14] C. D. Liu and J. Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[15] N. Lourenco and F. Pereira. PSO-CGO: A Particle Swarm Algorithm for Cluster Geometry Optimization. *International Journal of Natural Computing Research*, 2(1):1–20, 2011.

[16] P. M. Morse. Diatomic molecules according to the wave mechanics. ii. vibrational levels. *Physical Review*, 34(1):57–64, 1929.

[17] F. Pereira and J. Marques. A study on diversity for cluster geometry optimization. *Evolutionary Intelligence*, 2:121–140, 2009.

[18] K. Socha and M. Dorigo. Ant colony optimization for continuous domains. *European Journal of Operational Research*, 185(3):1155 – 1173, 2008.

[19] T. Stützle and H. H. Hoos. $\mathcal{MAX} - \mathcal{MIN}$ Ant system. *Future Generation Computer Systems*, 16(8):889 – 914, 2000.

[20] E. Taillard, P. Waelti, and J. Zuber. Few Statistical Tests for Proportions Comparison. *European Journal of Operational Research*, 185:1336–1350, 2008.

[21] S. Tsutsui. An enhanced aggregation pheromone system for real-parameter optimization in the aco metaphor. In *Ant Colony Optimization and Swarm Intelligence*, Lecture Notes in Computer Science 4150, pages 60–71. Springer, 2006.

[22] L. Wille and J. Vennik. Computational complexity of the ground-state determination of atomic clusters. *Journal of Physics A: Mathematical and General*, 18:419–422, 1985.

[23] A. Zaslavski and J. P. K. Doye. Physical perspectives on the global optimization of atomic clusters. In P. Pardalos and J. D. Pinter, editors, *Global Optimization*, volume 85 of *Nonconvex Optimization and Its Applications*, pages 103–139. Springer US, 2006.