Packing equal disks in a unit square: an immunological optimization approach

Giovanni Stracquadanio*, Ornella Greco†, Piero Conca‡, Vincenzo Cutello‡, Mario Pavone‡ and Giuseppe Nicosia‡

*Nuffield Department of Clinical Medicine
University of Oxford
OX3 9DU, Oxford, UK
Email: giovanni.stracquadanio@ludwig.ox.ac.uk

†Dept. of Mathematics, KTH
Royal Institute of Technology
10044, Stockholm, Sweden
Email: ogreco@kth.se

‡Dept. of Mathematics and Computer Science
University of Catania
95125, Catania, Italia
Email: pieroconca@gmail.com

***Abstract*—Packing equal disks in a unit square is a classical geometrical problem which arises in many industrial and scientific fields. Finding optimal solutions has been proved to be NP-hard, therefore, only local optimal solutions can be identified. We tackle this problem by means of the *optimization Immune Algorithm* (optIA), which has been proved to be among the best derivative-free optimization algorithms. In particular, OPTIA is used to pack up to 150 disks in a unit square. Experimental results show that the immune algorithm is able to locate the putative global optimum for all the instances. Moreover, a comparison with the *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) shows that OPTIA is more robust.**

I. INTRODUCTION

Packing items inside a container is one of the most studied geometrical optimization problem [1]. Particularly relevant is the problem of packing n non-overlapping identical disks of maximal radius r inside a unit square [2], [3]. This problem has received a lot of attention, since it finds application in fields like communication networks, facility location planning and molecular biology [4].

The problem is extremely complex from an optimization perspective, because it is characterized by reverse-convex constraints and a very large number of local minima. Deterministic methods have been successfully used to find optimal solutions along with proof of optimality for small instances. However, the computational cost of deterministic techniques increases considerably when packing a large number of disks is required. A different approach entails the use of stochastic methods, which have been shown to be a viable approach for large instances of this problem, although no optimality proof is provided in this case.

In particular, we followed this avenue by using the stochastic technique *optimization Immune Algorithm* (optIA) to tackle the problem of packing equal disks in a unit square. As a matter of fact, OPTIA is a population-based optimization method which mimics the process of clonal selection of the immune system of vertebrates. Furthermore, OPTIA has been shown to be among the best derivative-free optimizers in terms of effectiveness and efficiency [5].

Extensive simulations have been perfomed in order to assess the effectiveness of the approach being proposed. First, we compared OPTIA with the *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) for the problem of packing $n = 2, \ldots, 50$ disks: the experimental results show that OPTIA, as well as CMA-ES, is able to locate the putative global optimum, however, the immunological algorithm has higher robustness than CMA-ES. Successively, OPTIA is used to pack $n = 50, \ldots, 150$ disks; in this case the algorithm is able to locate the putative global optimum for all values, thus confirming its effectiveness also on large instances.

The rest of the paper is structured as follows. Section II provides a formal description of the problem of packing n circles in a unit square. Section III gives an overview of the different types of approaches that have been developed to deal with this problem. Section IV describes the pseudocode of OPTIA and the operators that govern its functioning. The results of the experiments, including the comparison between OPTIA and CMA-ES, are illustrated in Section V. The conclusions of this work are presented in Section VI.

II. DISK PACKING PROBLEMS

Packing disks inside a container is a classical problem in geometry and optimization [3]. In general, the container is a convex two-dimensional figure, such as a square, a circle or an equilateral triangle. Specifically, packing n equal disks in a unit square S requires finding the maximum radius r, such that the disks do not overlap and are fully contained in S.

Let $z_i = (x_i, y_i) \in \mathbb{R}^2$ be the center of i-th disk, and $C_r(z_i)$ the disk with center z_i and radius r. The problem of packing n disks in a unit square can be defined as follows:

$$\begin{array}{lll} \max r & s.t. & \\ & C_r(z_i) \subset S & i = 1, \ldots, n \\ & C_r^\circ(z_i) \cap C_r^\circ(z_j) = \emptyset & \forall i \neq j \end{array} \quad (1)$$

where C° denotes the interior of a disk.

The problem of finding the maximum common radius of identical and non overlapping disks in a unit square can be reduced to the problem of scattering n points in the unit square, such that the smallest pairwise distance is maximized. Specifically, by reducing the disk packing to a scattering problem, it is possible to define the following max-min problem:

$$r \quad = \quad \max \min_{i \neq j} \ \|z_i - z_j\|_2 \quad (2)$$

where r is the radius of the disks, and (z_i, z_j) are the centers of disks i and j, with $i, j = 1, \ldots, n$.

It has been showed that obtaining optimal solutions of problem 1 is equivalent to solving problem 2. Furthermore, using the formulation of problem 2, it is straightforward to prove that the problem becomes extremely difficult to solve because of quadratic reverse-convex constraints.

III. STATE-OF-THE-ART METHODS

The problem of packing disks in a square has been studied since 1965, when optimal packing strategies for up to 9 disks were found [6], [7]. This problem has been mainly tackled by two classes of methods: i) deterministic methods with optimality guarantee and ii) stochastic methods with no performance guarantee.

Deterministic methods have been used to prove optimality of packings for $n = 14$ [8], $n = 16$ [9], $n = 25$ [8], $n = 36$ [10]. Proof of optimality has been found for $n \leq 20$ [11], $21 \leq n \leq 27$ [12], $28 \leq n \leq 30$ [13].

Despite the ability to identify packings with optimality guarantee, deterministic methods become computationally increasingly expensive when used to solve large instances. For this reason, many stochastic optimization methods have been proposed: a basin hopping strategy was able to find optimal solutions for up to $n = 118$ [14], while a greedy search algorithm found 42 new putative optimal packing for $129 \leq n \leq 200$ [15]. Many putative optimal solutions for large packing instances of up to $n = 10,000$ disks have been found by Eckard Specht, using CSQ.

A complete list of optimal solutions are reported on Packomania[1].

IV. THE IMMULOGICAL OPTIMIZATION ALGORITHM

The class of Artificial Immune Systems (AIS) embodies a variety of algorithms inspired by the mechanisms of the immune system of biological creatures. In particular, the immune system has the function of detecting potentially harmful entities, called *Antigens* (Ag), and protecting the organism against them. A very powerful mechanism of the immune systems of vertebrates is the clonal selection principle, which allows to tailor the response of the immune system of such organisms to different and potentially previously unseen antigens. In particular, the immunological algorithms that mimic the mechanisms of the clonal selection principle are known as *Clonal Selection Algorithms* (CSA). Similarly to the way the clonal selection principle allows an immune system to adapt its response to a multitude of highly diverse antigens, the adaptivity of CSA makes these techniques a powerful tool for solving different types of problems. As a matter of fact, CSA have been used successfully to solve a large set of optimization problems [16], [17], [18], [19], [20], [21].

In this section, we propose the use of the *optimization Immune Algorithm* (OPTIA) to efficiently pack equal disks in a unit square. This *population-based* algorithm uses four operators: (i) cloning, (ii) inversely proportional hypermutation; (iii) hyper-macro mutation; and (iv) aging [5]. A candidate solution k is represented as a real vector $x^{(k)} = (x_1^{(k)}, \ldots, x_m^{(k)})$, where m is the dimension of the problem. The pseudo-code is presented in Fig. 1: first, OPTIA randomly generates an initial population $P^{(0)}$ containing d solutions and then it calculates the objective function value of each of these. Successively, OPTIA uses the *cloning* operator to generate dup clones of each solution, thus producing an intermediate population $P^{(clo)}$ of size $d \times dup$. At this step, each clone is assigned an *age* selected randomly in the interval $[0, \frac{2}{3}\tau_B]$, where τ_B is a parameter that controls its age and, therefore, determines the maximum number of iterations for which a solution is maintained in the population before being removed [19].

By mutating the solutions contained in $P^{(clo)}$, OPTIA is able to probe new regions of the search space, this capability is commonly refferred to as *exploration*. After cloning, in fact, OPTIA applies two mutation operators to the population of cloned solutions $P^{(clo)}$. In particular, we adopted the *inversely proportional hypermutation* operator and the *hyper-macromutation* operator based on the results presented in [22]. The inversely proportional hypermutation applies a convex mutation to each cloned solution in $P^{(clo)}$. In particular, let $i, j \in [1, m]$ be two distinct random integers, the i-th variable of the cloned solution k is perturbed according to the following expression:

$$x_i^{(k)} = (1 - \beta)x_i^{(k)} + \beta x_j^{(k)} \quad (3)$$

where $\beta \in [0, 1]$ is a random value. The hypermutation operator performs a number of mutations that is inversely proportional to the objective function value of the solution. Specifically, for each solution k, we defined a mutation potential $\alpha^{(k)}$ as follows:

$$\alpha^{(k)} = e^{-\rho \hat{f}(x^{(k)})}, \quad (4)$$

where $\rho \in [0, 1]$ is a parameter and $\hat{f}(x^{(k)})$ is the objective function value normalized in $[0, 1]$. Thus, the number of perturbed variables M for the solution k is determined as follows:

$$M^{(k)} = \lfloor m\alpha^{(k)} + 1 \rfloor \quad (5)$$

where m is the dimension of the problem. It is straightforward to note that in the case of the best solution only one mutation is performed. In this way, the better a solution is, the smaller the number of mutations that it undergoes

[1] www.packomania.com

procedure OPTIA($d, dup, \rho, \tau_B, T_{max}$)
  $t \leftarrow 0$;
  $FFE \leftarrow 0$;
  $P^{(t)} \leftarrow$ Initial_Population(d);
  Evaluate_fitness($P^{(t)}$);
  $FFE \leftarrow FFE + d$;
  $temp \leftarrow 0$
  while ($FFE < T_{max}$) **do**
    $P^{(clo)} \leftarrow$ Cloning ($P^{(t)}, dup$);
    $P^{(hyp)} \leftarrow$ Hypermutation($P^{(clo)}, \rho$);
    Evaluate_fitness ($P^{(hyp)}$);
    $FFE \leftarrow FFE + (d \times dup)$;
    $P^{(macro)} \leftarrow$ Hypermacromutation($P^{(clo)}$);
    Evaluate_fitness ($P^{(macro)}$);
    $FFE \leftarrow FFE + (d \times dup)$;
    $(P_a^{(t)}, P_a^{(hyp)}, P_a^{(macro)}) \leftarrow Aging(P^{(t)}, P^{(hyp)}, P^{(macro)}, \tau_B)$;
    $P^{(t+1)} \leftarrow (\mu + \lambda) -$ Selection($P_a^{(t)}, P_a^{(hyp)}, P_a^{(macro)}$);
    $t := t + 1$;
  end while
end procedure

Fig. 1. Pseudocode of OPTIA

and, therefore, the space sorrounding the solutions that appear to be more promising is investigated more thoroughly. This capability is also referred to as *exploitation*.

The hypermacromutation operator applies the convex perturbation (3) in a random fashion [23], [24]. Let $i, j \in [1, m]$ be two distinct random integers, for each solution k, the number of mutations $M^{(k)}$ is determined as follows:

$$M^{(k)} = |j - i + 1| \qquad (6)$$

where m is the dimension of the problem.

In general, the mutation operators perturb the population $P^{(clo)}$, generating the new populations $P^{(hyp)}$ and $P^{(macro)}$, respectively.

Successively, the algorithm uses the aging operator, it increases the age of the solutions in $P^{(hyp)}$, $P^{(macro)}$ and $P^{(t)}$ by one and discards all the solutions with age greater than τ_b. In our case, we adopted an *elitist aging operator*, which preserves the best solution found, in order to increase the convergence speed.

After applying the aging operator, the d best solutions from $P^{(hyp)}$, $P^{(macro)}$, and $P^{(t)}$ are selected to form the population $P^{(t+1)}$ for the next iteration. However, the aging operator can leave $\bar{d} < d$ solutions: in this case, $d - \bar{d}$ solutions are selected randomly from the solutions discarded by the aging operator in the previous iterations.

The algorithm stops when the number of *Fitness Function Evaluations* (FFE) reaches a predefined number T_{max}.

V. Experimental Results

The algorithms OPTIA is evaluated on two testbeds: 1) packing $n = 2, \ldots, 50$ equal disks and 2) packing $n = 51, \ldots, 150$ equal disks.

For the first testbed, 25 independent runs for each instance are performed, and the stopping criterion corresponds to the attainment of 5×10^6 fitness function evaluations. We compare OPTIA with the *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES). CMA-ES is an evolutionary algorithm that samples the search space by using a multi-variate Gaussian distribution. In particular, the algorithm adapts the covariance matrix of the distribution by taking into account the best solutions found at each iteration, using a classical (μ, λ)-selection scheme. This approach is aimed at maximizing the likelihood of sampling new points in the regions of the space in which the best solutions have been found.

The comparative technique CMA-ES is configured as follows: $\lambda = 5000$;, $\mu = 5000$ and default values are used for the remaining parameters, while OPTIA was tested by setting the population size to 200 and dup to 2. For each instance, we computed the mean (η) and the standard deviation (σ) over 25 runs, in order to assess the robustness of these methods.

Table I presents the results obtained by OPTIA and CMA-ES on the first testbed. The table shows that both algorithms are able to locate the putative global optimum for all the instances. In particular, OPTIA matches the best known solutions in all 25 runs for values of n up to 23. It should also be noted that the mean values across multiple runs are very close to the best known solutions for both algorithms. This result, in addition with the low values of standard deviation observed, confirm that both methods are consistently able to locate the basins of the best known solutions. However, by comparing the results of OPTIA with those of CMA-ES it appears that the immunological algorithm performs slightly better for increasing numbers of circles.

The results of the evaluation of OPTIA on the second testbed, with a number of circles $50 < n \leq 150$, show that also in this case OPTIA is able to find all the known best solutions for these instances. These results confirm that the immunological approach we are proposing is effective also on large scale problems.

TABLE I. Analysis of the solutions found by CMA-ES and optIA for the packing of $n = 1, \ldots, 50$ disks. For each instance, we report the putative global optimum, as reported in Packomania, along with the mean and standard deviation ($\eta \pm \sigma$) of the best individuals over 25 independent runs. Solutions equal to the putative global optimum are reported in bold face.

n	*best known*	CMA-ES *best*	CMA-ES $\eta \pm \sigma$	optIA *best*	optIA $\eta \pm \sigma$
2	0, 2928932188	**0, 2928932188**	0, 2928932188 ± 0, 0	**0, 2928932188**	0, 2928932188 ± 0, 0
3	0, 2543330950	**0, 2543330950**	0, 2543330950 ± 0, 0	**0, 2543330950**	0, 2543330950 ± 0, 0
4	0, 2500000000	**0, 2500000000**	0, 2500000000 ± 0, 0	**0, 2500000000**	0, 2500000000 ± 0, 0
5	0, 0207106781	**0, 0207106781**	0, 0196 ± 0, 001	**0, 0207106781**	0, 0207106781 ± 0, 0
6	0, 1876806011	**0, 1876806011**	0, 1876806011 ± 0, 0	**0, 1876806011**	0, 1876806011 ± 0, 0
7	0, 1744576302	**0, 1744576302**	0, 1744576302 ± 0, 0	**0, 1744576302**	0, 1744576302 ± 0, 0
8	0, 1705406887	**0, 1705406887**	0, 1694 ± 0, 001	**0, 1705406887**	0, 1705406887 ± 0, 0
9	0, 0166666667	**0, 1666666667**	0, 1666666667 ± 0, 0	**0, 1666666667**	0, 1666666667 ± 0, 0
10	0, 1482043226	**0, 1482043226**	0, 14795 ± 0, 00013	**0, 1482043226**	0, 1482043226 ± 0, 0
11	0, 1423992377	0,1423992373	0, 1425571231 ± 0, 0012	**0, 1423992377**	0, 1423992377 ± 0, 0
12	0, 1399588440	**0, 1399588440**	0, 13955 ± 0, 0011	**0, 1399588440**	0, 1399588440 ± 0, 0
13	0, 1339935135	**0, 1339935135**	0, 13256 ± 0, 0012	**0, 1339935135**	0, 1339935135 ± 0, 0
14	0, 1293317937	**0, 1293317937**	0, 129286 ± 0, 0007	**0, 1293317937**	0, 1293317937 ± 0, 0
15	0, 1271665475	**0, 1271665475**	0, 12695 ± 0, 0004	**0, 1271665475**	0, 1271665475 ± 0, 0
16	0, 1250000000	**0, 1250000000**	0, 1241986 ± 0, 00195	**0, 1250000000**	0, 1250000000 ± 0, 0
17	0, 1171967427	**0, 1171967427**	0, 1166 ± 0, 00065	**0, 1171967427**	0, 1171967427 ± 0, 0
18	0, 1155214325	**0, 1155214325**	0, 1143 ± 0, 0011	**0, 1155214325**	0, 1155214325 ± 0, 0
19	0, 1122654376	**0, 1122654376**	0, 11204 ± 0, 00042	**0, 1122654376**	0, 1122654376 ± 0, 0
20	0, 1113823475	**0, 1113823475**	0, 11065 ± 0, 0011	**0, 1113823475**	0, 1113823475 ± 0, 0
21	0, 1068602124	**0, 1068602124**	0, 10641 ± 0, 0004	**0, 1068602124**	0, 1068602124 ± 0, 0
22	0, 1056652968	**0, 1056652968**	0.10502 ± 0, 0016	**0, 1056652968**	0, 1056652968 ± 0, 0
23	0, 1028023234	**0, 1028023234**	0, 10220 ± 0, 000867	**0, 1028023234**	0, 1028023234 ± 0, 0
24	0, 1013818004	**0, 1013818004**	0, 10121456 ± 0, 244895	**0, 1013818004**	0.101379 ± 0, 00014
25	0, 1000000000	**0, 1000000000**	0, 0978 ± 0, 0011	**0, 1000000000**	0, 0989 ± 0, 00067
26	0, 0963623390	**0, 0963623390**	0, 09587 ± 0, 00055	**0, 0963623390**	0, 09599 ± 0, 00032
27	0, 0954200017	**0, 0954200017**	0, 09429 ± 0, 00087	**0, 0954200017**	0, 09489 ± 0, 00095
28	0, 0936728338	**0, 0936728338**	0, 09338 ± 0, 00095	**0, 0936728338**	0, 093578 ± 0, 00085
29	0, 0924631440	**0, 0924631440**	0, 09222 ± 0, 000356	**0, 0924631440**	0, 092387 ± 0, 00098
30	0, 0916710579	**0, 0916710579**	0, 09062 ± 0, 00095	**0, 0916710579**	0, 091539 ± 0, 00032
31	0, 0893383333	**0, 0893383333**	0, 0886 ± 0, 000347	**0, 0893383333**	0, 088516 ± 0, 00045
32	0, 0878581571	**0, 0878581571**	0, 0877 ± 0, 000558	**0, 0878581571**	0, 0876639 ± 0, 00346
33	0, 0872300141	**0, 0872300141**	0, 0856075 ± 0, 000215	**0, 0872300141**	0, 086294 ± 0, 00947
34	0, 0852703443	**0, 0852703443**	0, 08178 ± 0, 00961	**0, 0852703443**	0, 084296 ± 0, 00481
35	0, 0842907121	**0, 0842907121**	0, 083157 ± 0, 000567	**0, 0842907121**	0, 083745 ± 0, 00035
36	0, 0833333333	**0, 0833333333**	0, 082213 ± 0, 00027	**0, 0833333333**	0, 0829857 ± 0, 00582
37	0, 0820897664	**0, 0820897664**	0, 081155 ± 0, 0004587	**0, 0820897664**	0, 081749 ± 0, 00429
38	0, 0817009776	**0, 0817009776**	0, 08064 ± 0, 000778	**0, 0817009776**	0, 080849 ± 0, 00218
39	0, 0813675270	**0, 0813675270**	0, 07953 ± 0, 00062	**0, 0813675270**	0, 080034 ± 0, 00684
40	0, 0791867525	**0, 0791867525**	0, 078458 ± 0, 00042	**0, 0791867525**	0, 078995 ± 0, 0023
41	0, 0784502101	**0, 0784502101**	0, 07768 ± 0, 00048	**0, 0784502101**	0, 077954 ± 0, 0019
42	0, 0778015029	**0, 0778015029**	0, 07646 ± 0, 0006184	**0, 0778015029**	0, 076999 ± 0, 0029
43	0, 0763398106	**0, 0763398106**	0, 07518 ± 0, 00050765	**0, 0763398106**	0, 075897 ± 0, 0078
44	0, 0757819860	**0, 0757819860**	0, 07413 ± 0, 000475	**0, 0757819860**	0, 074897 ± 0, 0049
45	0, 0747273434	**0, 0747273434**	0, 07314 ± 0, 000755	**0, 0747273434**	0, 07369 ± 0, 0092
46	0, 0742721999	**0, 0742721999**	0, 0717996 ± 0, 00062	**0, 0742721999**	0, 073701 ± 0, 0024
47	0, 0731131513	**0, 0731131513**	0, 070675 ± 0, 000795	**0, 0731131513**	0, 07239107 ± 0, 0097
48	0, 07243229123	**0, 07243229123**	0, 06905 ± 0, 00171	**0, 07243229123**	0, 071782 ± 0, 0058
49	0, 0716926817	**0, 0716926817**	0, 0680107 ± 0, 00195	**0, 0716926817**	0, 070519 ± 0, 0064
50	0, 0713771038	**0, 0713771038**	0, 06576 ± 0, 002897	**0, 0713771038**	0, 0701326 ± 0, 0097

VI. Conclusions

Packing disks in a unit square is a complex geometrical optimization problem, which requires effective and efficient optimization methods to obtain satisfactory results. We introduced the *optimization Immunological Algorithm* (optIA), a population-based derivative free optimizer, which has been shown to be effective on many large scale optimization problems.

We used optIA for packing up to 150 disks; the experimental results showed that our approach is able to find the putative optimum for all instances. The comparison with CMA-ES shows that optIA performs slightly better and it is more robust.

References

[1] P. G. Szabó, M. C. Markót, and T. Csendes, "Global optimization in geometrycircle packing into the square," in *Essays and Surveys in Global Optimization*. Springer, 2005, pp. 233–265.

[2] D. W. Boll, J. Donovan, R. L. Graham, and B. D. Lubachevsky, "Improving dense packings of equal disks in a square," *the electronic journal of combinatorics*, vol. 7, no. 1, p. 46, 2000.

[3] M. Hifi and R. M'hallah, "A literature review on circle and sphere packing problems: models and methodologies," *Advances in Operations Research*, vol. 2009, 2009.

[4] I. Castillo, F. J. Kampas, and J. D. Pintér, "Solving circle packing problems by global optimization: numerical results and industrial applications," *European Journal of Operational Research*, vol. 191, no. 3, pp. 786–802, 2008.

[5] G. Nicosia, "Immune algorithms for optimization and protein structure prediction," Ph.D. dissertation, Department of Mathematics and Computer Science. University of Catania, 2004.

[6] J. Schaer, "The densest packing of nine circles in a square," *Canadian*

Mathematical Bulletin, vol. 8, pp. 273–277, 1965.

[7] J. Schaer and A. Meir, "On a geometric extremum problem," *Canadian Mathematical Bulletin*, vol. 8, no. 1, 1965.

[8] G. Wengerodt, "Die dichteste packung von 25 kreisen in einem quadrat," *Ann. Univ. Sci. Budapest Rolando E otv os Sect. Math*, vol. 30, pp. 3–15, 1987.

[9] ——, "Die dichteste packung von 16 kreisen in einem quadrat," *Beiträge zur Algebra und Geometrie*, vol. 16, pp. 173–190, 1983.

[10] K. Kirchner and G. Wengerodt, "Die dichteste packung von 36 kreisen in einem quadrat," *Beiträge zur Algebra und Geometrie*, vol. 25, pp. 147–159, 1987.

[11] R. Peikert, D. Würtz, M. Monagan, and C. de Groot, *Packing circles in a square: a review and new results*. Springer, 1992.

[12] K. J. Nurmela *et al.*, "More optimal packings of equal circles in a square," *Discrete & Computational Geometry*, vol. 22, no. 3, pp. 439–457, 1999.

[13] M. C. Markót and T. Csendes, "A new verified optimization technique for the" packing circles in a unit square" problems," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 193–219, 2005.

[14] B. Addis, M. Locatelli, and F. Schoen, "Packing circles in a square: New putative optima obtained via global optimization," *Optimization Online*, 2005.

[15] W. Huang and T. Ye, "Greedy vacancy search algorithm for packing equal circles in a square," *Operations Research Letters*, vol. 38, no. 5, pp. 378–382, 2010.

[16] V. Cutello and G. Nicosia, "Multiple learning using immune algorithms," *Proceedings of 4th International Conference on Recent Advances in Soft Computing, RASC*, pp. 102–107, 2002.

[17] A. Ciccazzo, P. Conca, G. Nicosia, and G. Stracquadanio, "An advanced clonal selection algorithm with ad-hoc network-based hypermutation operators for synthesis of topology and sizing of analog electrical circuits," in *Artificial Immune Systems*. Springer Berlin Heidelberg, 2008, pp. 60–70.

[18] V. Cutello and G. Nicosia, "A clonal selection algorithm for coloring, hitting set and satisfiability problems," in *Neural Nets*. Springer Berlin Heidelberg, 2006, pp. 324–337.

[19] M. Pavone, G. Narzisi, and G. Nicosia, "Clonal selection: an immunological algorithm for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 53, no. 4, pp. 769–808, 2012.

[20] V. Cutello, G. Narzisi, G. Nicosia, and M. Pavone, "Real coded clonal selection algorithm for global numerical optimization using a new inversely proportional hypermutation operator," *21st ACM SAC*, pp. 950–954, 2006.

[21] ——, "Clonal selection algorithms: a comparative case study using effective mutation potentials," in *Artificial Immune Systems*. Springer Berlin Heidelberg, 2005, pp. 13–28.

[22] V. Cutello, G. Nicosia, and M. Pavone, "Exploring the capability of immune algorithms: a characterization of hypermutation operators," in *Proceedings of the Third International Conference on Artificial Immune Systems (ICARIS'04)*, ser. LNCS, vol. 3239. Springer, 2004, pp. 263–276.

[23] V. Cutello, G. Nicosia, and N. Pavone, "An immune algorithm with hyper-macromutations for the dill's 2d hydrophobic-hydrophilic model," in *Proceedings of Congress on Evolutionary Computation (CEC'04)*, vol. 1. IEEE Press, 2004, pp. 1074–1080.

[24] N. Krasnogor, "Studies on the theory and design space of memetic algorithms," Ph.D. dissertation, University of the West of England, 2002.