Community of Scientist Optimization

An autonomy oriented approach to distributed optimization

Alfredo Milani [a,b] and Valentino Santucci [a]
[a] *Dept. of Mathematics and Computer Science*
University of Perugia
Perugia (Italy)
[b] *Department of Computer Science*
Hong Kong Baptist University
Hong Kong (China)
E-mail: {milani,valentino.santucci}@dmi.unipg.it

A novel optimization paradigm, called Community of Scientists Optimization (CoSO), is presented in this paper. The approach is inspired to the behaviour of a community of scientists interacting, pursuing for research results and foraging the funds needed to held their research activities. The CoSO metaphor can be applied to general optimization domains, where optimal solutions emerge from the collective behaviour of a distributed community of interacting autonomous entities. Experiments on benchmark problems in numerical optimization show the effectiveness of the approach. The CoSO framework presents some analogies with other evolutionary optimization approaches, and, with the design of appropriate foraging and competition strategies, it also represents a great potential for application in non-numerical and agent-based domains.
Keywords: Evolutionary Optimization, Autonomy Oriented Optimization, Numerical Optimization

1. Introduction

Computational solutions to hard optimization problems greatly benefit from the use of evolutionary techniques [1,14,15]. In particular, these techniques have been proven successful to tackle hard optimization problems such as the ones arising in continuous numerical [18,28] and combinatorial optimization domains [8,9]. Evolutionary techniques made use of various metaphors, often inspired by biological [7,20,23,27] or physical phenomena [5,21], in order to design heuristics and strategies which can be employed during the exploration of the solutions search space. Evolutionary Algorithms (EAs), in general, can be characterized as approximate methods which are based on some notion of time, e.g., generations or iterations, and some notions of autonomous computational entities [17], i.e., individuals, particles, ants, distributed agents, etc. The community of those entities encapsulates at some extent a (set of) partial solution(s). The entities act, interact and evolve over the time giving further approximations of the optimal solution. Different population dynamics [25] characterize the different approaches. In general the autonomous entities can reproduce themselves, breed, die, interact, change their characteristics and behaviour. The aim is to improve their local search performances and, eventually, collectively approach the optimal solution.

In recent years, biological behaviours [7,18, 23] have inspired a number of computational metaphors for EAs. The underlying hypothesis is to exploit the fact, which has been observed in many biological environments, that the emerging behaviour of a number of simple distributed agents (such as bees, ants, schooling fishes, birds, etc.) can exhibit a high level of organization and high level valuable properties such as converging to some (near) optimal solution.

The idea of Community of Scientist Optimization (CoSO) originates from the investigation of the mechanism of the collective emerging behaviour of a very interesting biological organization: the human scientific community. CoSO can be seen as a distributed optimization model where computational autonomous entities are governed by rules which exploit the mechanisms that humans employ in order to organize, select and finance the scientific research, while obtaining as a collective product the advancement of scientific knowledge. Despite of the suggestive inspiration, the goal of CoSO is to investigate the effectiveness and the actual applicability of those mechanisms to computational optimization problems.

AI Communications
ISSN 0921-7126, IOS Press. All rights reserved

The rest of the paper is organized as follows. Next section analyzes the main features of the modern scientific research process seen as a collective emergent behaviour of a community of autonomous entities. Section 3 introduces the CoSO approach and its evolutionary foraging optimization model applied in the framework of numerical optimization. In Section 4, relations between CoSO and other metaheuristics in literature are investigated. Experimental results of a CoSO implementation in numerical benchmark problems and comparisons with Particle Swarm Optimization (PSO) [18] and Bacterial Foraging Optimization (BFO) [23] performances are discussed in Section 5. Finally, conclusions are drawn in Section 6 where future lines of research are also depicted.

2. Research Process as Emerging Behaviour

2.1. Science from Patronage to Self Organization

Since the remote times of great Greek mathematicians, scientists and philosophers, until the Renaissance and beyond, the scientific research has been pursued mostly by relying on the goodwill of a *mecenate* or patron. The *mecenate* was usually a prince, a king or a very rich person, who sponsored the activity of a "recognized scientist", often by paying him/her a regular salary, and admitting him/her to its court in exchange of a different variety of services from art performances to scientific talks (this was the case, for example, of Leonardo Da Vinci with the court of Ludwig the Moor in Milan, and of the Greek scientist Archimede, who was applying his scientific findings to the design of a parabolic burning mirror weapon for the Greek navy against the Romans). The ability of *the mecenate* in selecting the right scientists and, conversely, the ability of the scientist to cope with the personal idiosyncrasies of the patron, were crucial in the development of science of these early times.

It was with the first scientific societies of 1600 and eventually with the advent of the industrial revolution that the process of scientific production started to boost, with the huge production of the XIX century and the never ceasing amount of high valuable findings and results continuously produced.

It must be noted that the most relevant change which took place in scientific production process was the moving from a top-down driven process, based on the arbitrary graceful judgment of the patron, into a distributed self organized system which self regulates its own expansion and evaluation criteria. Relevant elements in the modern scientific research are the role of scientific journals and committees in selecting papers to publish in the journals (i.e., blind peer reviewing), and the more objective criteria, (e.g., impact factor, citation index, etc.) adopted by government agencies to assign research funds. After a preliminary verification of the consistency and sustainability of project proposals, *funds assignment criteria* are usually based: 1) on the coherence with strategic objectives and balancing between areas of research, usually established by the government, and 2) on the evaluation of publications and previous results of the proponents. The foraging mechanism induced by the funds management produces an interesting emergent behaviour on the scientific community. The community of scientists tend to *explore* all the research areas, and to *exploit* the best results in each area. With the peer-reviewing mechanism it is the community itself which indirectly assesses the projects to finance.

2.2. Features and Emerging Behaviour of the Research Process

There are some concepts and mechanisms which regulate the evolution of the scientific research process which are worth to be briefly discussed and which will be later integrated in the CoSO model:

- *Scientists / Researchers*: scientists are autonomous actors doing research by interacting with other scientists and using different resources. Sometimes they are organized in hierarchies, i.e., chief scientists, project investigators, research assistants, etc.
- *Funds*: scientists need resources to do their research and are financially supported by funds that they employ to hire new researchers or to buy tools, labs, books, etc.
- *Journals*: journals are collections of results which act as *communication channels* among scientists. Scientists read journals: to take inspiration from previous researches, to avoid to rediscover already known results and to improve previous results.

- *Results*: scientific results are findings of the scientists which they submit to journals for publications.
- *Selection / Publication*: the scientific production is *peer-reviewed*. Reputable scientists select the papers that other scientists want to publish, i.e., to draw to public attention. The idea underlying peer-reviewing is that papers evaluation is held by mean of objective criteria which anyone else can agree on. A result is usually selected for publication in two cases: 1) when it is new and it improves a previous one, 2) when it informs that a certain hypothesis is false or a certain research line is not promising (e.g., the Gödel incompleteness theorems that gave a negative answer to Hilbert's second problem so stopping the research on this topic).
- *Research projects*: research investigators submit, to funding subjects (e.g., governmental agencies), their research projects, i.e., proposals containing description of the research area and plans of research detailing which resources are needed and how to employ them. A typical research management strategy dilemma is deciding whether to hire new researchers with short time contracts or to devote existing researchers for a longer time to the project.
- *Funds assignment criteria and policies*: the funds are assigned upon projects and are based on the scientific results of the proponents, i.e., as a rule of thumb, scientific groups with the best results are more likely to obtain funds. Moreover, governmental agencies also guarantee that additional (and possibly conflicting) criteria are met, such as prioritizing strategic research topics and ensuring *research topics diversity*. For instance, the European Union enforces topics diversity by funding a limited number of *outsiders* or *off-the-track* projects per year; a strict requirement for these high risk projects is that they must concern challenging, new or underexplored areas.

The scientific research process is probably the most notable example of collective intelligence where the valuable emergent behaviour is represented by the advancement of knowledge. In the scientific community, each researcher autonomously interacts with the others by reading journals (communication) and producing new results either by exploring new directions of research (exploration) or by deepening existing lines of research (exploitation). In both cases if the results improve the previous ones (selection), the new ideas can be published and spread in the community thus representing an inspiration for further researches. Successful researches will more likely lead to obtain funds to continue the research (foraging), while non successful ones mean no funds and the end of the research activity. The journals, together with the foraging mechanism of funds, introduce a competition among scientists which indirectly acts as a selection mechanism. Once funds have been obtained, the successful proponent has to decide a *strategy of research funds management* which consists in deciding the *resources distribution* and the *research direction*. In a *resource distribution policy* the proponent has to decide if and how many new researchers to hire, how much resources allocate to them, or if it just keeps doing research by himself with more resources, e.g., for a longer time. Decisions about *research directions* regulate where and how the new and the old researchers should explore, i.e., they should focus in the same area (exploitation) or they should enlarge the perspective area (exploration). As mentioned before, *funds distribution policies* are also important as a global regulatory mechanism. Indeed, governmental agencies can establish that certain areas are strategic, or that certain areas of research cannot go below a minimal amount of resources, or that too many projects insist on the same area. Policies which aim at topics diversity can be seen as general heuristics which guarantee a balanced advancement of knowledge, and redistribute the risk of failure when research projects are too much dense in an area and local optima can be found.

3. Community of Scientist Optimization: an Evolutionary Foraging Optimization Model

CoSO is an evolutionary autonomy oriented optimization algorithm whose key features are inspired from the metaphor of the scientific research process taking place in a community of scientists. The emergent collective behaviour in CoSO is suitable for numerical optimization problems. CoSO employs concepts such as autonomous entities, local communication, foraging, selection, reproduc-

tion, local search, randomized exploration and others, which can be individually found in many Evolutionary Algorithms (EAs), and can be ultimately observed and explained in the framework of the collective process of scientific research.

Let a multidimensional numerical optimization problem be represented by an objective function $f : \Theta \to \mathbb{R}$ to be minimized (or maximized) in the space of feasible solutions $\Theta \subseteq \mathbb{R}^d$ (where d is a positive integer representing the dimensionality of the problem), CoSO consists of a dynamic set of autonomous entities, *researchers*, $R = \{r_1, \ldots, r_{RN}\}$ that share one or more journals $J_j \in J = \{J_1, \ldots, J_{JN}\}$ and compete for publishing their best results, i.e., the best points visited in the search space Θ with respect to the cost/fitness function f. Researchers use funds to organize research by also hiring new researchers to help them. At each iteration a researcher consumes one unit of funds, thus the researchers can possibly die by funds exhaustion. The activities of searching, publishing, funds distribution, and funds investment are synchronized by discrete time instants, also called iterations. As the iterations progress, the journals will reflect the advancement of knowledge on function f eventually converging toward the optimal minimum (or maximum) value.

The CoSO general scheme is formally presented in Algorithm 1, while the algorithm steps and the entities involved are detailed in the following.

Algorithm 1 CoSO General Scheme

```
procedure CoSO
    t ← 0
    initialize()
    while not termination criterion do
        t ← t + 1
        funds ← 0
        for each researcher r_i do
            v_{i,t} ← changeDir(v_{i,t−1}, b_{i,t−1}, J)
            x_{i,t} ← moveTo(x_{i,t−1}, v_{i,t})
            f(x_{i,t}) ← eval(x_{i,t})
            m_{i,t} ← m_{i,t−1} − 1
            funds ← funds + 1
            b_{i,t} ← updateCurr(x_{i,t}, b_{i,t−1})
            addToGlobalReport(f(x_{i,t}), x_{i,t})
            submit(f(x_{i,t}), x_{i,t})
        end for
        for each journal J_j do
            updateJournal(J_j)
        end for
        assignFunds(funds, globalReport)
        for each researcher r_i do
            if m_{i,t} = 0 then
                die(r_i)
            else
                hire(r_i)
            end if
        end for
    end while
end procedure
```

3.1. Researchers

Researchers represent the active autonomous entities of CoSO, which actually perform the search in Θ. At each time instant t, a researcher r_i is characterized by a certain set of properties:

- $x_{i,t}$, a d-dimensional vector indicating the *research position* in the multidimensional space Θ at time t,
- $v_{i,t}$, a d-dimensional vector indicating the *direction of research* with respect to the previous position at time step $t-1$,
- $b_{i,t}$, a d-dimensional vector indicating the *personal curriculum* and representing the personal best result, i.e., the best position ever visited until time step t by researcher r_i,
- $\{\rho_{i,1}, \ldots, \rho_{i,JN}\}$, a probability distribution on the set of journals J (see below) describing the probability to publish in each journal other than the weights that the researcher gives to each journal in the reading phase,
- $m_{i,t}$, a non-negative integer amount of *funds*,
- $s_i \in [0, 1]$, a *fund management strategy*.

At each iteration, researchers autonomously perform several actions:

- *die*, when $m_{i,t} = 0$ the researcher is removed from set R while its *results* remain recorded in the journals. Existing researchers can die by funds exhaustion after some unsuccessful iterations in which case they cease their activity in CoSO. The death by funds exhaustion acts as the selection rule of classical Genetic Algorithms (GAs) [22] and as a distributed regulatory rule of researchers population size.
- *hire*, one or more new researchers can be created by the current researcher provided it has enough funds. This results in new autonomous entities to be added to R (see Section 3.4).
- *changeDir*, the researcher changes its current direction of research $v_{i,t}$ depending on its own

results other than the recent results published in the journals (see below).
- *moveTo*, the researcher moves to a new position x_t to explore in the search space according to its own research direction v_t (see below).
- *eval*, the researcher computes the objective / fitness value $f(x_{i,t})$, i.e., the result of exploring the point x_t of the search space.
- *updateCurr*, in the case that the new result improves personal knowledge, the researcher curriculum is updated according to $b_{i,t} \leftarrow \arg\min\{f(x_{i,t}), f(b_{i,t-1})\}$ (for minimization problems).
- *submit*, the new result $(f(x_{i,t}), x_{i,t})$ is *submitted* for evaluation/publication to issue t of journal J_s chosen among the journals in J by means of a roulette wheel tournament. In this case, $\rho_{i,j}$ represents the probability of journal J_j to be chosen for submission by researcher r_i (see Section 3.2 for the journal acceptance policy).

The two procedures that implement the "directed navigation" of each researchers, i.e., *changeDir* and *moveTo*, are detailed in the following.

3.1.1. changeDir

The new direction of research is computed according to:

$$v_{i,t} \leftarrow \omega v_{i,t-1} + \varphi_1 \beta_{1,t}(b_{i,t-1} - x_{i,t-1}) + \\ + \varphi_2 \beta_{2,t} \left(\sum_{j=1}^{JN} \rho_{i,j}(J_{j,t-1,c_j} - x_{i,t-1}) \right) \quad (1)$$

where the weights consist in the inertia ω, the acceleration factors φ_1, φ_2, and the random factors $\beta_{1,t}, \beta_{2,t}$ which are uniformly distributed in $[0, 1]$. Factors $\rho_{i,j}$ are the coefficients of a convex combination, i.e., $\forall j \in [1, JN] \cap \mathbb{N} : \rho_{i,j} \geq 0 \wedge \sum_j \rho_{i,j} = 1$. Moreover, the three terms in rule (1) characterize the local behaviour of the researcher. The first term, called the inertia or momentum, serves as a memory of the previous *direction of research* preventing the researcher from drastically changes direction. The second term, the *cognitive component*, models the tendency of the researcher to move towards the direction of its personal curriculum $b_{i,t-1}$. The third term, the *social component*, tries to direct the search towards the latest results in journals. In the expression $\rho_{i,j}(J_{j,t-1,c_j} - x_{i,t-1})$ each parameter $\rho_{i,j}$ of the convex combination represents the significance of the journal issue $J_{j,t-1}$ for the researcher r_i, while $J_{j,t-1,c_j}$ is the component c_j of issue $t-1$ of journal J_j, where c_j is an integer randomly chosen in $[1, |J_j|]$[1]. Note that similar cognitive and social structures appear also in the PSO framework [18], as discussed in Section 4.

3.1.2. moveTo

At each iteration a researcher with non-zero funds, i.e., $m_{i,t} > 0$, moves to explore another search position in Θ depending on its current direction of research $v_{i,t}$ and according to:

$$x_{i,t} \leftarrow x_{i,t-1} + v_{i,t} \quad (2)$$

3.2. Journals

CoSO journals $\{J_j\}$ are a set of data structures which record the significant progress of exploration done by the researchers over the time. A journal J_j is formally characterized by:

- a *journal length* k_j, i.e., the maximum number of results which can be published in a journal issue,
- a sequence of *journal issues* $\{J_{j,t}\}$, one issue for each discrete time instant t.

A journal issue $J_{j,t}$ is a list of at most k_j papers, i.e., pairs $(f(x_{i,u}), x_{i,u})$ (where i ranges on the researchers set R and $u \in [0, t] \cap \mathbb{N}$) ordered with respect to $f(x_{i,u})$, which contains (at most) the best k_j results obtained until iteration t by the journal publishers. Researchers will refer to the latest journal issues they read in order to decide their direction of research (see rule (1)). Researchers will also publish in the journals they know (see Section 3.1). In this sense journals act as communication channels among researchers.

The collection of results composing a journal is incrementally built by using the *acceptance policy* defined as follows. A result submitted at time t to a journal J_j is published in the journal issue $J_{j,t}$ if and only if it is within the best k_j results submitted to J_j of all the times, i.e., it should improve the best previously published results.

Finally note that, although the acceptance policy aforementioned is the one used in our experi-

[1]Note that $|J_j|$ indicates the length, i.e., the number of papers, of journal J_j.

ments, also other schemes can be formulated like, for example, a *crowding* approach aiming at maintaining a certain (genotypic or phenotypic) diversity in the papers set.

3.3. Funds Assignment Policy

Different policies can be defined for funds distribution. In CoSO, at each iteration, the funds amount reassigned equals the amount of funds consumed in the previous iteration. This results in a *closed economy model* where the total amount of funds possessed by all the researchers at a given time is constant, i.e., $\forall t_1, t_2 : \sum_i m_{i,t_1} = \sum_j m_{j,t_2}$ where i (j) sums over the researchers existing at time t_1 (t_2). Anyway, since new researchers can be created by successful ones, and old researchers with no funds can disappear, the total number of researchers and the amount of available funds per researcher can dynamically vary throughout the evolution.

Moreover, the distribution of funds among the researchers can follow different criteria. In the following we introduce the distribution criterion used in this work and that can be considered as composed by two components: a basic foraging criterion, and a dynamic research topics diversity criterion. Both criteria employs randomized techniques which ensure the aim of the criterion asymptotically rather then deterministically.

3.3.1. Basic Foraging Criterion

The basic criterion for funds assignment consists in prizing the researcher results obtained during the last iteration. However, a percentage $\Omega_t \in [0, 1]$ of the total available funds is also reserved to outsiders, i.e., to completely new researcher positions. As described below, the value of Ω_t is dynamically self-regulated during the iterations.

The performances obtained by all researchers in the last iteration are recorded in a *global report*. The global report is the ordered list of all results at time t and it is the basis for funds distribution.

Let $F = |R|$ the total amount of funds available at time t, where R is the dynamic set of researchers at time $t-1$[2], and let i the position of a researcher in the global report, then: for each *unit* of funds $F - \lceil F \cdot \Omega_t \rceil$ to assign, a random roulette wheel tournament is made and researcher in position i in the global report receives a probability $\frac{|R|-i+1}{|R|\cdot(|R|+1)/2}$ to obtain the fund *unit*. Thus, the probability is proportional to the position scored by the researcher in the global report.

The randomness of this method do not prevent any researcher from receiving funds and the criterion also verifies two likely properties: it reflects the performances ranking, and it gives a relatively high difference among first and last results. Moreover, the use of a roulette wheel tournament based on ranking positions, instead of on fitness values, allows the foraging criterion to scale up to different granularities of the fitness landscape. Indeed, conversely from the fitness based tournament, it allows to differentiate the researchers probabilities to receive funds also in the case where many fitness values in the global report are similar.

3.3.2. Dynamic Research Topics Diversity: Outsiders

A policy for topics diversity aims at maintaining diversity in the researchers population in order to avoid stagnation, i.e., a premature convergence of all researchers toward the same local optimum.

Diversity in CoSO is maintained by financing projects of outsiders, i.e., hiring researchers in new random areas. A self regulated mechanism has been devised to allow a low level of diversity when the community of researchers is not converging and, conversely, a higher level in the case of convergence. The percentage of outsiders funds $\Omega_t \in [\Omega_{min}, \Omega_{max}]$ varies between the given bounds, and it is initially assigned to $\Omega_0 \leftarrow \Omega_{min} + \frac{\Omega_{max}-\Omega_{min}}{2}$.

The standard deviation σ_t of the fitness values in the global report at time t is a suitable index of convergence of the population and it is used to update Ω_t according to:

$$\Omega_t \leftarrow \begin{cases} \Omega_{t-1} + \frac{\Omega_{max}-\Omega_{min}}{2} \cdot \epsilon_\Omega^+ \text{ if } \sigma_t < \sigma_0 \\ \Omega_{t-1} - \frac{\Omega_{max}-\Omega_{min}}{2} \cdot \epsilon_\Omega^- \text{ otherwise} \end{cases} \quad (3)$$

where ϵ_Ω^+ and ϵ_Ω^- are the increment/decrement constants.

It is to be noticed that the standard deviations σ_t are computed by using the fitness values of the last population of researchers. Although genotypic distances seem to be in principle more appropriate than fitness values, it has experimentally observed that the use of the latter indicators provides anyway a good diversity measure with the additional property of a more efficient computation.

[2] $|R|$ is the cardinality of set R.

The amount $\lceil F \cdot \Omega_t \rceil$ of outsiders funds is then assigned to hire a number of outsiders. The number of those new outsiders is determined with the same method described in the next section for the number of new hired researchers, while the initial properties values of each new outsider are randomly assigned as in the initialization phase (see Section 3.5).

3.4. Funds Management Strategy: hire *action*

The *hire* action is strictly connected with the funds management strategy. Each researcher who has an amount of funds greater than one unit (either just received or maintained from previous iterations) has to decide whether to keep them or to invest them for hiring new researchers. In this latter case it is necessary to decide how many researchers and how much resources assign to them. This decision is randomly regulated by the strategy parameter $s_i \in [0, 1]$ which characterizes each researcher behaviour according to the pseudo-code in Algorithm 2.

Algorithm 2 Fund Management Strategy Pseudo-code

```
personalFunds ← m_{i,t} + newAssignment
if personalFunds > 0 then
    m_{i,t} ← ⌈personalFunds · s_i⌉
    newFunds ← personalFunds − m_{i,t}
    newRes ← randomInt(1, newFunds)
    createAndInitialize(newRes, newFunds)
end if
```

In other words the researcher keeps from itself a percentage s_i of funds units and uses the rest to hire a random number of researchers.

When, at time t, a new researcher r_{new} is created, it receives an initial amount of funds $m_{new,t} \leftarrow newFunds/newRes$ from its creator / supervisor r_i, while it almost completely inherits the other relevant properties:

- the personal curriculum $b_{i,t}$ of its creator / supervisor r_i,
- a random direction of research $v_{new,t}$, where, for each dimension j, $v_{new,t,j} \leftarrow N(0, \epsilon_v)$, with $N(0, \epsilon_v)$ representing a random number generated following a Gaussian distribution with mean 0 and standard deviation ϵ_v, i.e., the new researcher moves toward a random direction with a small initial momentum ϵ_v,
- a position $x_{new,t} \leftarrow x_{i,t} + v_{new,t}$, obtained by moving from its supervisor r_i in the multi-dimensional space Θ by one step at velocity $v_{new,t}$,
- a fund management strategy $s_{new} \leftarrow N(s_i, \epsilon_s)$ which is obtained by a Gaussian perturbation of the supervisor strategy, and where ϵ_s is a small global constant,
- a distribution of relevance $\{\rho_{new}\}$ on the journals set J obtained by normalizing the result of the Gaussian perturbation $\rho_{new,j} \leftarrow N(\rho_{i,j}, \epsilon_\rho)$, where ϵ_ρ is a small constant value.

It is worth noticing that the new hired researchers are spread in a contour of the supervisor position, thus realizing an *exploitation strategy*, i.e., a sort of *local search* in promising areas of the search space. In this way, the hiring mechanism described allows the search dynamic of each researcher to mix the two canonical search styles:

- *by depth*, where each researcher/supervisor proceeds in the search by updating its research direction as usual,
- *by breadth*, obtained by hiring new researchers that initially move in the surround of the supervisor.

3.5. Coso Initialization

Each CoSO configuration is characterized by an initial number of researchers RN_0 and the total amount of funds TF. Thus, at the begin, RN_0 researchers are deployed, each one with an initial amount of fund units $m_{i,0} \leftarrow TF/RN_0$. Moreover, uniformly random values are assigned for positions (x), research directions (v), fund management strategies (s), and probability distributions on the journals set (ρ).

Instead, the JN journals, each one with length k_j, are empty at the begin and they will be progressively filled up with the results submitted/selected as described in Section 3.2.

3.6. CoSO Guidelines for Parameters Choices

CoSO requires to choose various parameters values that altogether affect the diversity rate of the population thus the exploration / exploitation balance of the algorithm.

The initial population size RN_0 and the total amount of funds TF regulate the complexity

of the algorithm. Larger values for these parameters clearly increase the computational requirements per iteration. However, a large amount of researchers per iteration can allow CoSO to better explore the problem search space and to increase the ability of escape from local optima. The dynamically varying population size, differently from other EAs using a fixed population size, allows CoSO to dispose an adequate number of computational resources for each particular iteration, basing on the researchers experience accumulated during the evolution. Moreover, it is worthwhile to note that TF constitutes an upper bound for the population size and, as rule of thumb and as done in our experiments (see Section 5), we suggest to set $RN_0 \leftarrow TF/5$.

The number of available journals JN indirectly influences the potential diversity of the population due to the social behaviour of the researchers. Since journals act as a memory for the more attractive solutions visited so far, a larger JN allows to maintain a more diverse set of attractive solutions. Moreover, although in a smaller way, the same consideration can be made for the journal lengths k_j.

The diversity control parameters Ω_{min}, Ω_{max}, ϵ_Ω^+, ϵ_Ω^- influence the diversity rate of CoSO population by specifying the diversity bounds and the diversity step of variation. Anyway, the use of a self-regulated diversity control parameter Ω_t, together with the comparison value σ_0 derived from the initial randomly generated population (see Section 3.3.2), thus unaffected from algorithm biases, allows to set these parameters to some constant values that result largely independent from the problem at hand. As done in our experiments (see Section 5), we suggest to use the following setting: $\Omega_{min} = 0.2$, $\Omega_{max} = 0.5$, $\epsilon_\Omega^+ = 0.2$, $\epsilon_\Omega^- = 0.1$.

The perturbation values ϵ_v, ϵ_s, ϵ_ρ act in a similar way of the perturbation parameters of the popular Evolution Strategy (ES) algorithm [2]. Each one of these parameters should be set to some small value in order to exploit the parent property that led to good results and, at the same time, to slightly vary the parent behaviour in order to try to improve it. Anyway, it is necessary a distinction. Since ϵ_s and ϵ_ρ are adopted to perturb probability values, they are independent from the fitness landscape at hand and can be set to some constant values independently from the problem instance to optimize (in Section 5 we adopt and suggest $\epsilon_s = \epsilon_\rho = 0.1$[3]). However, ϵ_v is largely dependent on the granularity or scale of the particular fitness landscape at hand, thus it should be adequately chosen for each problem instance or it can be dinamically regulated during the evolution through some automatic regulatory mechanism (however, this is beyond the scope of this paper).

Finally, in the direction update rule (1), the parameters ω, φ_1, φ_2 respectively regulate, for each researcher, the tendency to follow the previous research direction, the tendency to exploit the personal knowledge accumulated, and the tendency to swarm, i.e., to follow the best results found by the entire community. Moreover, these parameters have a similar meaning of those present in the PSO scheme [18], for which a theoretical-based setting ($\omega = 0.7298$, $\varphi_1 = \varphi_2 = 1.49618$) has been suggested in [6,24].

4. Properties of CoSO and Related Works

The CoSO framework integrates concepts such as autonomous distributed entities, local communication, foraging, selection, reproduction, local search, randomized exploration and others, which can be individually found in many Evolutionary Algorithms (EAs).

CoSO shares various elements with PSO [18,24], and more in general it subsumes this latter one. Indeed, since an easy parallel can be done between the notion of CoSO researchers and PSO particles, a global PSO scheme (with all particles connected in a full network) can be modeled by CoSO by using a single journal of length 1 and a funds distribution strategy which reassigns at each iteration one unit of funds to each researcher. In this way no researchers are created or deleted.

On the other hand many important differences with classical EAs exist, first of all the introduction of inheritance, foraging and selection mechanisms completely absent in the pure PSO approaches.

It is interesting to recognize some basic mechanisms of foraging: survival and indirect communications (see for instance indirect communication through pheromone in ACO [7]). Journals act as

[3] Due to Gaussian distributions properties, in this way we have that about the 99.7% of the perturbations result in no more than 0.3 "points of probability".

communication channels that researchers use to indirectly exchange information about where areas showing good results are, i.e., where the food for surviving is. Funds are computational resources which are guaranteed to the best computing entities, i.e., the best researchers. Connecting the foraging to the performances, by the funds distribution strategy and allowing communication through channels/journals makes possible to obtain a collective emergent behaviour consisting in optimizing the performances, i.e., a collective convergence to a (near) optimal solution.

CoSO also uses elements from classical Genetic Algorithms (GAs) [10,22,27]. The foraging mechanism induced by the notion of research funds introduces a selection mechanism which resembles the genetic survival of the fittest strategy [22]. In other words, CoSO implements a kind of "publish or perish" rule which can be restated in a "get good results then get funds or stop activity" rule. Also the inheritance methods used in CoSO can be certainly related to GAs, whereas the foraging selection mechanism allows to promote the best researcher properties. A remarkable difference is that asexual self-reproduction (a kind of *parthenogenesis*) is used in CoSO, i.e., hired researchers tend to reflect most of the features of their creator/supervisor, or to evolve them by mutation (see the perturbation schemes in Section 3.4). With this respect CoSO can be related also to bacteria foraging hybrid algorithms as [3,19,20].

Furthermore, other hybrid approaches can be related to CoSO, for instance [26] which introduces diversity control in a Particle Swarm optimizer, or the recent [25,29] where population dynamics and genetic operators are used in the PSO framework.

Although the many connections with proposed hybrid approaches, CoSO has the remarkable merit of discovering some similar mechanisms within the unique coherent metaphor of *scientific production process*.

Finally, another interesting aspect of CoSO is that, despite its application in the domain of numerical optimization, it can be easily extended to other areas and used as a framework for managing distributed agents in problems suitable to be solved by collective emergent behaviour. Consider, for instance, applications domains where agents (i.e., retrieval agents crawling for "interesting" documents, planning agents, web services, etc.) produce non-numerical solution instances or services, which can be compared and shared through journals. The agents can have, in general, different computation capabilities/abilities which will be prized with different computational resources by the foraging mechanism.

5. Experiments

5.1. Experiments Design

The performances of CoSO have been evaluated on a suite of fourteen well known numerical benchmark functions proposed in [4] and reported in Table 1. This suite presents a variegate combination of problems. Functions $f_1, \ldots, f_3$ are simple unimodal problems, $f_4, \ldots, f_9$ are multimodal problems with many local minima, while $f_{10}, \ldots, f_{14}$ are higly complex, but low dimensional, multimodal problems. Furthermore, since the initialization interval, reported in Table 1 for each benchmark, constitutes a subspace of the entire feasible region that does not contain the known global optimum (see [4]), it is avoided being deceived from possible algorithmic biases toward the area of the space in which the population is initialized [4,12].

Each benchmark is investigated with dimensionality $d = 10$ and $d = 30$, apart $f_{10}, \ldots, f_{14}$ that present a given dimensionality, i.e., $d = 2$ for f_{10}, f_{11} and $d = 4$ for $f_{12}, \ldots, f_{14}$. For the convenience of the following description we call "experiment" the combination of benchmark and dimensionality, thus having a total of 23 experiments.

In a preliminary phase, non-systematic tests with different initial parameters combinations have been conducted, so we have concluded to choose the following parameters setting for more systematic tests and comparisons.

For each experiment, CoSO has been executed with three different values for the journal number parameter JN, i.e., 1, 3, and 5. Each journal has length $k = 10$. The diversity control constants have been set to $\Omega_{min} = 0.2$, $\Omega_{max} = 0.5$, $\epsilon_{\Omega}^{+} = 0.2$, $\epsilon_{\Omega}^{-} = 0.1$, while the other perturbation constants are $\epsilon_v = 0.01$, $\epsilon_s = \epsilon_\rho = 0.1$.

Moreover, for comparison purposes, also a basic Bacterial Foraging Optimization (BFO) algorithm [23] and the PSO schems with a global (gPSO) and a ring (rPSO) network topology [4,18] have been tested. In order to perform a more significant comparison, both for CoSO and for PSO schemes, the

same parameters values for the direction/velocity update rule have been employed, i.e. $\omega = 0.7298$, $\varphi_1 = \varphi_2 = 1.49618$ (note that these values are the ones discussed in Section 3.5 and theoretically derived, for PSO, in [6,24]). The CoSO initial number of researchers RN_0 and the total amount of funds TF have been set to $RN_0 = 30$ and $TF = 150$ (i.e. 5 initial funds units per researcher), while the PSO swarm size has been set to 40. The choice of a greater initial population size for PSO is justifiable by the fact that CoSO, conversely from PSO, dynamically varies the population size throughout the evolution and it has been observed that it usually tends to grow up. Instead, for BFO, a setting similar to the one used in [13] has been adopted, i.e., 40 bacteria, 50 chemotactic steps, 4 swim steps, 5 reproduction steps, 3 and 7 elimination and dispersal steps (depending on the allowed cap of fitness evaluations), 0.25 as probability of elimination, 0.1 as run length unit.

In order to introduce more confidence on the statistical results, for each benchmark and for each algorithm 50 executions have been held. For each run the allowed cap of fitness evaluations (NFES) has been set to $T = 10\,000 \cdot d$ for the benchmarks $f_1, \ldots, f_9$ and to $T = 300\,000$ for the other benchmarks[4].

Performances were measured as the minimum error $f(x) - f(x^{opt})$ found at the end of the run ($f(x^{opt})$ is the known global minimum fitness value for the problem). Moreover an execution is regarded convergent if $f(x) - f(x^{opt}) \leq \epsilon$, where ϵ has been set to 10^{-2}. Therefore, measures of convergence frequency and convergence speed were obtained by recording the number of convergent executions and the NFES needed to converge.

5.2. Experimental Results

Table 2 shows a qualitative comparison among the tested algorithms. Indeed, the fitness average and standard deviation computed on the 50 executions are reported for each algorithm and for each benchmark. These results show that, in general, CoSO is able to reach more qualitative solutions than PSO and BFO, and that a small number of journals is preferable on those problems that present a relatively high dimensionality.

For completeness, in Table 3, a statistical comparison, relative to the final fitness values (see Table 2) between CoSO with $JN = 3$ and rPSO is reported. This comparison has been done using the modified Bonferroni procedure [16] consisting in a classical t-test where the α values are manipulated in a way that allows to increase the confidence on the results obtained [4]. Substantially, the p-values of each experiment are recorded and ranked from smallest to largest, these ranks are then inverted, and the α value for each experiment is computed by dividing an initial α (0.05 in our case) for the inverse-rank of the experiment. The comparison shows that the large majority of performance differences, apart the equally performed cases, are statistically significant.

In Table 4 the success rate SR (i.e., the number of convergent executions above the total number of executions) and the average NFES of all convergent executions C_{avg} are reported. These two indices are also synthesized in the quality measure $Q_m = C_{avg}/SR$ introduced in [11]. The results clearly show that CoSO outperforms PSO and BFO schemes both in convergence speed that in robustness. Furthermore, with a small JN the convergence speed is speeded up, although sometimes the algorithm does not converge to a solution with an enough quality. Conversely, employing a greater number of journals allows to confer a stronger robustness to CoSO scheme. This seems also to be coherent with the intuition that a small number of journals allows an exploitative behaviour, while, increasing JN allows to confer more exploration to CoSO.

Finally, in Figure 1 a convergence graph averaged over all the 23 experiments is reported. In order to weigh the experiments in a more equal as possible way, we have normalized the average fitness values within the interval $[0, 100]$, so we have computed their average across the various benchmark functions, and then plotted them on the convergence graph reported. From this overall view, it can be seen more clearly that $JN = 3$ represents a good setting for the journals number parameter, and that CoSO converges faster, other than to better solutions, with respect to PSO and BFO schemes.

[4]Note that, in the BFO case, in order to remain in the allowed cap of NFES, two different numbers of elimination and dispersal steps are adopted, respectively, 3 when $T = 100\,000$ and 7 when $T = 300\,000$. Moreover, each BFO execution is stopped after the allowed number of fitness evaluations has been performed.

6. Conclusion

CoSO is an innovative evolutionary approach to computational optimization based on the distributed autonomous mechanisms used by the scientific community to manage the process of scientific production. Among the main relevant features of CoSO there are: the use of the researchers, i.e., a community of distributed and interacting computational entities, a *foraging mechanism*, i.e., a competition for research funds which indirectly acts as a selection mechanism, a self regulating criterion *"outsider" strategy* which ensures to maintain a certain degree of diversity in the research topics (i.e., coverage and exploration with respect to focusing and exploitation due to the foraging), and evolving *research management strategies* which dynamically adapt the population size by creating new computational entities.

Despite of the many points of contacts with recent hybrid PSO, GAs and various foraging proposals [3,20,23,25], the CoSO metaphor offers a single framework where foraging, competition, communication, and search dynamics lead to a collective emergent behaviour which results in an efficient optimization process.

Experimental results in numerical optimization problems are encouraging since CoSO outperforms classical PSO and BFO schemes [18,23] in some standard benchmark problems.

Future lines of research will regards: the exploration of different criteria for funds assignments (e.g., taking into account the historical performances of researchers), different evolution mechanism for journal relevance distribution and for funds management strategy (which currently do not evolve in the single researcher but in its outbreeds), different selection schemes for journals (e.g., taking into account also the diversity degree of each journal issue other than its quality), and the use of some self-adaptive techniques for the algorithm parameters.

Finally, an interesting line of research will be also the experimentation of CoSO as framework for organizing the collective behaviour of distributed set of agents in the area of non-numerical problems.

References

[1] Bäck T. (1996) Evolutionary Algorithms in Theory and Practice, Oxford, NY

[2] Beyer H. G., Schwefel H. P. (2002) Evolution strategies–A comprehensive introduction. In: Natural Computing, 1(1):3–52, Springer

[3] Biswas A., Dasgupta S., Das S., Abraham A. (2007) Synergy of PSO and Bacterial Foraging Optimization – A Comparative Study on Numerical Benchmarks, In: Innovations in Hybrid Intelligent Systems, ASC 44, pp. 255–263, Springer

[4] Bratton D, Kennedy J (2007) Defining a Standard for Particle Swarm Optimization. In: Swarm Intelligence Symposium, 2007, SIS 2007. IEEE, Honolulu, HI, pp. 120-127

[5] Cerny V. (1985) A Thermodynamical Approach to the Travelling Salesman Problem: an Efficient Simulation Algorithm. In: Journal of Optimization Theory and Applications, 45:41–51

[6] Clerc M., Kennedy J. (2002) The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space. In: IEEE Transactions on Evolutionary Computation, 6(1):58–73

[7] Colorni A., Dorigo M., Maniezzo V. (1991) Distributed Optimization by Ant Colonies. In: Proceedings of First European Conference on Artificial Life, Elsevier Publishing, pp. 134–142

[8] Dorigo M., Gambardella L. M. (1997) Ant Colony System : A Cooperative Learning Approach to the Traveling Salesman Problem. In: IEEE Transactions on Evolutionary Computation, 1(1):53–66

[9] Dorigo M., Maniezzo V., Colorni A. (1996) Ant System: Optimization by a Colony of Cooperating Agents. In: IEEE Transactions on Systems, Man, and Cybernetics – Part B, 26(1):29–41

[10] Eiben A. E., Raué P. E., Ruttkay Z. S. (1994) Genetic Algorithms with Multi-Parent Recombination. In: Proceedings of Third Conference on Parallel Problem Solving from Nature, pp. 78–87

[11] Feoktistov V. (2006) Differential Evolution. In search of solutions. Springer

[12] Gehlhaar D. K., Fogel D. B. (1996) Tuning evolutionary programming for conformationally exible molecular docking. Evolutionary Programming, pp. 419-429

[13] Hanning C., Yunlong Z., Kunyuan H. (2011) Adaptive Bacterial Foraging Optimization. In: Abstract and Applied Analysis, Hindawi Publishing Corporation

[14] Hingston P. F., Barone L. C., Michalewicz Z. (2008) Design by Evolution: Advances in Evolutionary Design. Springer

[15] Holland J. H. (1975) Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor

[16] Jaccard J., Wan C. K. (1996) LISREL approaches to interaction effects in multiple regression. Sage Publications, Thousand Oaks, CA

[17] Liu J., Jin X., Tsui K. (1995) Autonomy-Oriented Computing: From Problem Solving to Complex Systems Modeling. Kluwer Academic Publishers/Springer, Boston, 2005.

[18] Kennedy J., Eberarth R. (1995) Particle Swarm Optimization. In: Proceedings of IEEE Conference on Neural Networks, IEEE Press, pp. 1942–1948

[19] Kim D. H., Abraham A., Cho J. H. (2007) A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Global Optimization. In: Information Sciences, 177(18):3918–3937

[20] Kim D. H., Cho C. H. (2005) Bacterial Foraging Based Neural Network Fuzzy Learning. In: Proceedings of Indian International Conference on Artificial Ingelligence, pp. 2030–2036

[21] Kirkpatrick S., Gelatt C. D., Vecchi M. P. (1983) Optimization by Simulated Annealing. In: Science New Series 220(4598):671–680

[22] Michalewicz Z. (1999) Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag

[23] Passino K.M. (2002) Biomimicry of bacterial foraging for distributed optimization and control. In: IEEE Control Systems Magazine, vol. 22, no. 3, pp. 52–67.

[24] Poli R., Kennedy J., Blackwell T. (2007) Particle Swarm Optimization. An overview. In: Swarm Intelligence 1(1):33–57

[25] Qi K., Lei W., Qi-Di W. (2008) A Novel Ecological Particle Swarm Optimization Algorithm and its Population Dynamics Analysis. In: Applied Mathematics and Computation 205(1):61–72

[26] Riget J., Vesterstrøm J. S. (2002) A Diversity-Guided Particle Swarm Optimizer – the ARPSO. In: EVALife Technical Report no. 2002-02

[27] Schmitt L. M. (2001) Theory of Genetic Algorithms. In: Theoretical Computer Science, 259:1–61

[28] Storn R., Price K. (1997) Differential evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. In: Journal of Global Optimization, 11(4):341–359

[29] Yanjiang M., Zhihua C., Jianchao Z. (2009) Dynamic Population-Based Particle Swarm Optimization Combined with Crossover Operator. In: Proceedings of Ninth International Conference on Hybrid Intelligent Systems, vol. 1, pp. 399–404

Table 1
Benchmark Functions

Function Definition	Feasible Bounds	Initialization Bounds
$f_1(x) = \sum_{i=1}^{d} x_i^2$	$[-100, 100]^d$	$[50, 100]^d$
$f_2(x) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} x_j\right)^2$	$[-100, 100]^d$	$[50, 100]^d$
$f_3(x) = \sum_{i=1}^{d-1} \left\{100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right\}$	$[-30, 30]^d$	$[15, 30]^d$
$f_4(x) = -\sum_{i=1}^{d} x_i \sin(\sqrt{x_i})$	$[-500, 500]^d$	$[-500, -250]^d$
$f_5(x) = \sum_{i=1}^{d} \left\{x_i^2 - 10\cos(2\pi x_i) + 10\right\}$	$[-5.12, 5.12]^d$	$[2.56, 5.12]^d$
$f_6(x) = -20\exp\left\{-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right\} - \exp\left\{\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)\right\} + 20 + e$	$[-32, 32]^d$	$[16, 32]^d$
$f_7(x) = \sum_{i=1}^{d} x_i^2/4000 - \prod_{i=1}^{d}\cos(x_i/\sqrt{i}) + 1$	$[-600, 600]^d$	$[300, 600]^d$
$f_8(x) = \frac{\pi}{d}\left\{10\sin^2(\pi y_1) + \sum_{i=1}^{d-1}(y_i - 1)^2\left\{1 + 10\sin^2(\pi y_{i+1})\right\} + (y_d - 1)^2\right\} +$	$[-50, 50]^d$	$[25, 50]^d$
$+ \sum_{i=1}^{d} \mu(x_i, 10, 100, 4)$		
$y_i = 1 + (x_i + 1)/4$		
$\mu(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & \text{if } x_i > a \\ 0 & \text{if } -a \le x_i \le a \\ k(-x_i - a)^m & \text{if } x_i < -a \end{cases}$		
$f_9(x) = 0.1\left\{\sin^2(\pi x_1) + \sum_{i=1}^{d-1}(x_i - 1)^2\left\{1 + \sin^2(3\pi x_{i+1})\right\} +\right.$	$[-50, 50]^d$	$[25, 50]^d$
$\left.+(x_d - 1)^2\left\{1 + \sin^2(2\pi x_d)\right\}\right\} + \sum_{i=1}^{d}\mu(x_i, 5, 100, 4)$		
$f_{10}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	$[-2, 2]^2$	$[1, 2]^2$
$f_{11}(x) = \left\{1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right\} \times$	$[-5, 5]^2$	$[2.5, 5]^2$
$\times \left\{30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right\}$		
$f_{12}(x) = -\sum_{i=1}^{5}\left\{\sum_{j=1}^{4}(x_j - a_{ij})^2 + c_i\right\}^{-1}$	$[0, 10]^4$	$[7.5, 10]^4$
$f_{13}(x) = -\sum_{i=1}^{7}\left\{\sum_{j=1}^{4}(x_j - a_{ij})^2 + c_i\right\}^{-1}$	$[0, 10]^4$	$[7.5, 10]^4$
$f_{14}(x) = -\sum_{i=1}^{10}\left\{\sum_{j=1}^{4}(x_j - a_{ij})^2 + c_i\right\}^{-1}$	$[0, 10]^4$	$[7.5, 10]^4$

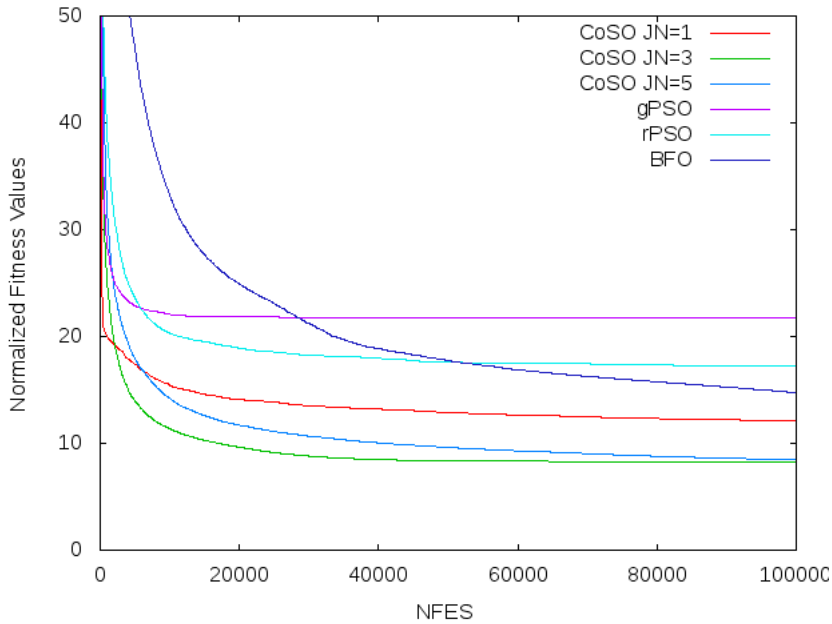

Fig. 1. Normalized Convergence Graph

Table 2
Experimental Results: Fitness Values

f	CoSO JN=1	CoSO JN=3	CoSO JN=5	gPSO	rPSO	BFO
$d = 10$ — $T = 100\,000$						
f_1	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	0.0005 (±0.0001)
f_2	**0** (±0)	**0** (±0)	5.54e–06 (±6.75e–06)	**0** (±0)	**0** (±0)	0.0352 (±0.0098)
f_3	8.0199 (±0.3557)	2.8244 (±2.7556)	**0.7540** (±1.0936)	441.31 (±967.84)	3.5986 (±4.7164)	17.09 (±4.3568)
f_4	880.58 (±87.49)	**0** (±0)	**0** (±0)	1386.27 (±129.89)	1070.68 (±62.49)	1184.38 (±0.0002)
f_5	**0** (±0)	**0** (±0)	**0** (±0)	31.50 (±19.26)	8.0039 (±4.0157)	21.30 (±5.0639)
f_6	**0** (±0)	**0** (±0)	**0** (±0)	17.94 (±5.4106)	12.53 (±9.4009)	0.7554 (±3.5016)
f_7	**0** (±0)	0.0565 (±0.0213)	0.0184 (±0.0180)	0.0794 (±0.0330)	0.0362 (±0.0245)	0.0095 (±0.0025)
f_8	0.0397 (±0.0095)	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	1.51e–05 (±4.79e–06)
f_9	0.1764 (±0.0354)	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	5.13e–05 (±1.36e–05)
$d = 30$ — $T = 300\,000$						
f_1	**0** (±0)	**0** (±0)	2.40e–07 (±2.44e–07)	19333.33 (±11234.87)	**0** (±0)	**0** (±0)
f_2	**0** (±0)	1.8899 (±1.1620)	1106.68 (±380.06)	73227.37 (±51704.90)	334.15 (±1247.04)	14.24 (±16.07)
f_3	28.29 (±0.2613)	37.24 (±29.67)	49.57 (±28.62)	3.73e+07 (±4.94e+07)	**13.75** (±8.2176)	99.44 (±1.6044)
f_4	6465.17 (±201.64)	**0** (±0)	0.0004 (±0.0009)	4756.38 (±394.07)	4618.24 (±210.84)	3553.15 (±1.36e–12)
f_5	**0** (±0)	**0** (±0)	2.0473 (±1.4773)	250.96 (±41.18)	181.74 (±32.02)	121.74 (±5.2789)
f_6	**0** (±0)	**0** (±0)	0.0002 (±0.0001)	19.85 (±0.0527)	19.79 (±0.0980)	8.9869 (±6.8283)
f_7	**0** (±0)	0.0130 (±0.0124)	0.0013 (±0.0036)	183.49 (±91.64)	**0** (±0)	**0** (±0)
f_8	0.2811 (±0.0295)	**0** (±0)	**0** (±0)	5.12e+07 (±1.22e+08)	**0** (±0)	11.45 (±0.5578)
f_9	1.9802 (±0.0711)	**0** (±0)	2.84e–07 (±2.44e–07)	8.20e+07 (±1.64e+08)	**0** (±0)	**0** (±0)
d various — $T = 300\,000$						
f_{10}	9.29e–08 (±1.33e–07)	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)
f_{11}	0.0168 (±0.0252)	**0** (±0)	**0** (±0)	48.60 (±39.68)	**0** (±0)	81.00 (±0)
f_{12}	**4.9243** (±0.6331)	5.0524 (±1.78e–15)	5.0524 (±1.78e–15)	5.0524 (±0)	5.0524 (±0)	5.0524 (±1.78e–15)
f_{13}	5.3155 (±0.0232)	**5.2740** (±8.88e–16)	**5.2740** (±8.88e–16)	**5.2740** (±2.66e–15)	**5.2740** (±2.66e–15)	**5.2740** (±8.88e–16)
f_{14}	**5.3491** (±0.1700)	5.3606 (±8.88e–16)	5.3606 (±8.88e–16)	5.3606 (±2.66e–15)	5.3606 (±2.66e–15)	5.3606 (±8.88e–16)

Table 3

Statistic Significance for CoSO with $JN = 3$ vs rPSO

Experiment	**p-value**	**Inverse-Rank**	**α-value**	**Significant**
f_4 $(d = 10)$	0	23	0.002174	yes
f_5 $(d = 10)$	0	22	0.002273	yes
f_6 $(d = 10)$	0	21	0.002381	yes
f_7 $(d = 10)$	0	20	0.0025	yes
f_4 $(d = 30)$	0	19	0.002632	yes
f_5 $(d = 30)$	0	18	0.002778	yes
f_6 $(d = 30)$	0	17	0.002941	yes
f_3 $(d = 30)$	0.000001	16	0.003125	yes
f_7 $(d = 30)$	0.000006	15	0.003333	yes
f_2 $(d = 30)$	0.04226	14	0.003571	no
f_3 $(d = 10)$	0.04932	13	0.003846	no
f_1 $(d = 10)$	1	12	0.004167	no
f_2 $(d = 10)$	1	11	0.004545	no
f_8 $(d = 10)$	1	10	0.005	no
f_9 $(d = 10)$	1	9	0.005556	no
f_1 $(d = 30)$	1	8	0.00625	no
f_8 $(d = 30)$	1	7	0.007143	no
f_9 $(d = 30)$	1	6	0.008333	no
f_{10}	1	5	0.01	no
f_{11}	1	4	0.0125	no
f_{12}	1	3	0.016667	no
f_{13}	1	2	0.025	no
f_{14}	1	1	0.05	no

Table 4

Experimental Results: Success Rates and Convergence Speed

f	**CoSO JN=1**			**CoSO JN=3**			**CoSO JN=5**			**gPSO**			**rPSO**			**BFO**		
	SR	**C_{avg}**	**Q_m**	**SR**	**C_{avg}**	**Q_m**	**SR**	**C_{avg}**	**Q_m**	**SR**	**C_{avg}**	**Q_m**	**SR**	**C_{avg}**	**Q_m**	**SR**	**C_{avg}**	**Q_m**
$d = 10$ — $T = 100\,000$																		
f_1	1.00	361	**361**	1.00	5368	5368	1.00	15527	15527	1.00	5026	5026	1.00	10428	10428	1.00	78495	78495
f_2	1.00	444	**444**	1.00	13882	13882	1.00	58828	58828	0.82	9035	11018	1.00	25776	25776	0.00	—	—
f_3	0.00	—	—	0.04	32300	807499	0.16	49400	**308749**	0.00	—	—	0.02	24100	1204999	0.00	—	—
f_4	0.00	—	—	1.00	11537	**11537**	1.00	28015	28015	0.00	—	—	0.00	—	—	0.00	—	—
f_5	1.00	299	**299**	1.00	15422	15422	1.00	38270	38270	0.00	—	—	0.00	—	—	0.00	—	—
f_6	1.00	356	**356**	1.00	7232	7232	1.00	18999	18999	0.08	8900	111249	0.46	26643	57919	0.00	—	—
f_7	1.00	394	**394**	0.02	10200	509999	0.32	40628	126962	0.02	15100	754999	0.10	30340	303399	0.58	97770	168568
f_8	0.00	—	—	1.00	4710	4710	1.00	12382	12382	0.98	3854	**3932**	1.00	10254	10254	1.00	54603	54603
f_9	0.00	—	—	1.00	6764	6764	1.00	15063	15063	1.00	4512	**4512**	1.00	10189	10189	1.00	60256	60256
$d = 30$ — $T = 300\,000$																		
f_1	1.00	457	**457**	1.00	45791	45791	1.00	154452	154452	0.12	15575	129791	1.00	34844	34844	1.00	143294	143294
f_2	1.00	843	**843**	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—
f_3	0.00	—	—	0.02	116000	**5799999**	0.00	—	—	0.02	280150	14007499	0.00	—	—	0.00	—	—
f_4	0.00	—	—	1.00	86985	**86985**	1.00	241134	241134	0.00	—	—	0.00	—	—	0.00	—	—
f_5	1.00	397	**397**	0.98	167574	170993	0.02	254800	12739999	0.00	—	—	0.00	—	—	0.00	—	—
f_6	1.00	381	**381**	1.00	54351	54351	1.00	183008	183008	0.00	—	—	0.00	—	—	0.32	227746	711706
f_7	1.00	396	**396**	0.52	51900	99807	0.96	172847	180048	0.06	16283	271383	0.98	42210	43071	1.00	150308	150308
f_8	0.00	—	—	1.00	42713	**42713**	1.00	115116	115116	0.46	23926	52013	1.00	58273	58273	0.00	—	—
f_9	0.00	—	—	1.00	79628	79628	0.98	162444	165759	0.44	18672	**42436**	1.00	45086	45086	1.00	119067	119067
d various — $T = 300\,000$																		
f_{10}	1.00	672	672	1.00	424	**424**	1.00	967	967	1.00	633	633	1.00	686	686	1.00	7876	7876
f_{11}	0.64	15268	23856	1.00	1124	**1124**	1.00	2681	2681	0.46	2291	4980	0.88	2165	2460	0.00	—	—
f_{12}	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—
f_{13}	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—
f_{14}	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—	0.00	—	—