Chaos, Solitons and Fractals 110 (2018) 259–266

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Mimicking the collective intelligence of human groups as an optimization tool for complex problems

Ilario De Vincenzo[a], Giovanni F. Massari[a], Ilaria Giannoccaro[a], Giuseppe Carbone[a,b,c,*], Paolo Grigolini[d]

[a] Department of Mechanics, Mathematics and Management, Politecnico di Bari, v.le Japigia 182, Bari 70126, Italy
[b] Physics Department M. Merlin, CNR Institute for Photonics and Nanotechnologies U.O.S. Bari via Amendola 173, Bari 70126, Italy
[c] Department of Mechanical Engineering, Imperial College London, London, South Kensington Campus, London SW7 2AZ, United Kingdom
[d] Center for Nonlinear Science, University of North Texas, P. O. Box 311427, Denton, Texas 76203-1427, USA

ARTICLE INFO

Article history:
Received 23 October 2017
Revised 16 March 2018
Accepted 26 March 2018
Available online 4 April 2018

Keywords:
Optimization algorithm
Artificial intelligence
Collaborative decisions
Decision making
Group decision
Social interactions
Complexity
Markov chains

ABSTRACT

A large number of optimization algorithms have been developed by researchers to solve a variety of complex problems in operations management area. We present a novel optimization algorithm belonging to the class of swarm intelligence optimization methods. The algorithm mimics the decision making process of human groups and exploits the dynamics of such a process as a tool for complex combinatorial problems. In order to achieve this aim, we employ a properly modified version of a recently published decision making model [64,65], to model how humans in a group modify their opinions driven by self-interest and consensus seeking. The dynamics of such a system is governed by three parameters: (i) the reduced temperature βJ, (ii) the self-confidence of each agent β', (iii) the cognitive level $0 \le p \le 1$ of each agent. Depending on the value of the aforementioned parameters a critical phase transition may occur, which triggers the emergence of a superior collective intelligence of the population. Our algorithm exploits such peculiar state of the system to propose a novel tool for discrete combinatorial optimization problems. The benchmark suite consists of the NK - Kauffman complex landscape, with various sizes and complexities, which is chosen as an exemplar case of classical NP-complete optimization problem.

A comparison with genetic algorithms (GA), simulated annealing (SA) as well as with a multiagent version of SA is presented in terms of efficacy in finding optimal solutions. In all cases our method outperforms the others, particularly in presence of limited knowledge of the agent.

1. Introduction

Human groups are proven to outperform single individuals in solving a variety of complex tasks in many different fields, including new product development, organizational design, strategy planning, research and development. Their superior ability originates from the collective decision making: individuals make choices, pursuing their individual goals on the basis of their own knowledge/expertise and adapting their behavior to the actions of the other agents. Social interactions, indeed, promote a mechanism of consensus seeking within the group, but also provide a useful tool for knowledge and information sharing [1–4,40]. This type of decision making dynamics is common to many social systems in Nature, e.g., flocks of birds, herds of animals, ant colonies, school of fish [40–50], as well as bacterial colonies [5–7], and even to artificial systems [8–11].

Even though the single agent possesses a limited knowledge, and the actions it performs are usually very simple, the collective behavior leads to the emergence of a superior intelligence known as swarm or collective intelligence [12–15,29], which in the last years have seen a huge growth of applications in the field of optimization swarm-based algorithms in operations management context [30–33]. The swarm algorithms exploit the collective intelligence of the social groups, such as flock of birds, ant colonies, and schools of fish, in accomplishing different tasks. They include the Ant Colony Optimization (ACO) [17–19], the Particle Swarm Optimization [20], the Differential Evolution [21], the Artificial Bee Colony [22,23], the Glowworm Swarm Optimization [24,25], the Cuckoo Search Algorithm [26], and very recently the

* Corresponding author.
E-mail address: carbone@poliba.it (G. Carbone).

https://doi.org/10.1016/j.chaos.2018.03.030

Grey Wolf Optimizer [27] and the Ant Lion Optimizer [28]. These algorithms share remarkable features, such as decentralization, self-organization, autonomy, flexibility, and robustness, which have been proven very useful to solve complex operational tasks [34,35]. Applications of ACO algorithm mainly concern the traveling salesman problem, scheduling, vehicle routing, and sequential ordering [36]. More recently, they have been also employed in supply chain contexts to solve production-inventory problems [37,38] and network design [39].

In this paper we propose a novel swarm intelligence optimization algorithm to solve complex combinatorial problems. The proposed algorithm is inspired by the behavior of human groups and their ability to solve a very large variety of complex problems, even when the individuals may be characterized by cognitive limitations. Although it is widely recognized that human groups, such as organizational teams, outperform single individuals in solving many different tasks including new product development, R&D activities, production and marketing issues, literature is still lacking of optimization algorithms inspired by the problem solving process of human groups. Similarly to other social groups, human groups are collectively able, by exploiting the potential of social interactions, to achieve much better performance than single individuals can do. This specific ability of human groups has been defined as group collective intelligence [51,52] that recently is receiving a growing attention in the literature as to its antecedents and proper measures [51,52].

The proposed algorithm, hereafter referred to as Human Group Optimization (HGO) algorithm, is developed within the methodological framework recently proposed by CG [53,54] to model the collective decision making of human groups. This model captures the main drivers of the individual behavior in groups, i.e., self-interest and consensus seeking, leading to the emergence of collective intelligence. The group is conceived as a set of individuals making choices based on rational calculation and self-interested motivations. However, any decision made by the individual is also influenced by the social relationships he/she has with the other group members. This social influence pushes the individual to modify the choice he/she made, for the natural tendency of humans to seek consensus and avoid conflict with people they interact with [55]. As a consequence, effective group decisions spontaneously emerge as the result of the choices of multiple interacting individuals.

To test the ability of HGO algorithm, we compare its performance with those of some benchmarks chosen among trajectory-based and population-based algorithms. In particular, the HGO is compared with the Simulated Annealing (SA), a Multi Agent version of the Simulated Annealing (MASA) and with genetic algorithms (GA).

2. The decision making model of human groups

Here we briefly summarize the decision making model presented in Ref. [53,54]. We consider a human group made of M socially interacting members, which is assigned to accomplish a complex task. The task is modelled in terms of N binary decisions and the problem consists in solving a combinatorial decision making problem by identifying the set of choices (configuration) with the highest fitness, out of 2^N configurations.

As an example of application of the method, the fitness landscape, i.e., the map of all configurations and associated fitness values, is generated following the classical NK procedure (see Appendix A for more details), where N are the decisions and K the interactions among them. Each decision d_i of the vector $\mathbf{d}$ is a binary variable $d_i = \pm 1,\ i = 1, 2, \ldots, N$. Each vector $\mathbf{d}$ is associated with a certain fitness value $V(\mathbf{d})$ computed as the weighted sum of N stochastic contributions $W_j\left(d_j, d_1^j, d_2^j, \ldots, d_K^j\right)$ that each decision leads to the total fitness. The contributions $W_j\left(d_j, d_1^j, d_2^j, \ldots, d_K^j\right)$ depend on the value of the decision d_j itself and the values of other K decisions $d_i^j,\ i = 1, 2, \ldots, K$, and are determined following the classical NK procedure [56–58]. The fitness function is then defined as

$$V(\mathbf{d}) = \frac{1}{N}\sum_{j=1}^{N} W_j\left(d_j, d_1^j, d_2^j, \ldots, d_K^j\right) \tag{1}$$

The integer index $K = 0, 1, 2, \ldots, N-1$ corresponds to the number of interacting decision variables, and tunes the complexity of the problem: increasing K increases the complexity of the problem. Individuals are characterized by cognitive limits, i.e. they posses a limited knowledge. The level of knowledge of the kth member of the group is identified by the parameter $p \in [0,\ 1]$, which is the probability that each single member knows the contribution of the decision to the total fitness.

Based on the level of knowledge, each member k computes his/her own perceived fitness (self-interest) as follows:

$$V_k(\mathbf{d}) = \frac{\sum_{j=1}^{N} D_{kj} W_j\left(d_j, d_1^j, d_2^j, \ldots, d_K^j\right)}{\sum_{j=1}^{N} D_{kj}}. \tag{2}$$

where $\mathbf{D}$ is the matrix whose elements D_{kj} take the value 1 with probability p and 0 probability $1 - p$.

During the decision making process, each member of the group makes his/her choices to improve the perceived fitness (self-interest) and to seek consensus within the group. The dynamics is modelled by means of a continuos-time Markov process where the state vector $\mathbf{s}$ of the system has $M \times N$ components $\mathbf{s} = (s_1, s_2, \ldots, s_n) = \left(\sigma_1^1, \sigma_1^2, \ldots \sigma_1^N, \sigma_2^1, \sigma_2^2, \ldots \sigma_2^N, \ldots, \sigma_M^1, \sigma_M^2, \ldots \sigma_M^N\right)$. The variable $\sigma_k^j = \pm 1$ is a binary variable representing the opinion of the member k on the decision j. The probability $P(\mathbf{s}, t)$ that at time t, the state vector takes the value $\mathbf{s}$ out of 2^N possible states, satisfies the master equation

$$\begin{aligned}\frac{dP}{dt} = &-\sum_l w\left(\boldsymbol{s}_l \to \boldsymbol{s}_l'\right) P(\boldsymbol{s}_l, t) \\ &+ \sum_l w\left(\boldsymbol{s}_l' \to \boldsymbol{s}_l\right) P\left(\boldsymbol{s}_l', t\right)\end{aligned} \tag{3}$$

where $\boldsymbol{s}_l = (s_1, s_2, .., s_l., s_n)$ and $\boldsymbol{s}_l' = (s_1, s_2, ., -s_l., s_n)$. The transition rate of the Markov chain (i.e. the probability per unit time that the opinion s_l flips to $-s_l$ while the others remain temporarily fixed) is defined so as to be the product of the transition rate of the Ising–Glauber dynamics [59], which models the process of consensus seeking to minimize the conflict level, and the Weidlich exponential rate [60,61], which models the self-interest behavior of the agents:

$$\begin{aligned}w\left(\boldsymbol{s}_l \to \boldsymbol{s}_l'\right) = &\frac{1}{2}\left[1 - s_l \tanh\left(\beta \frac{J}{< \kappa >} \sum_h A_{lh} s_h\right)\right] \\ &\times \exp\left\{\beta'\left[\Delta V\left(\boldsymbol{s}_l', \boldsymbol{s}_l\right)\right]\right\}\end{aligned} \tag{4}$$

In Eq. (4) A_{lh} are the elements of the adjacency matrix, $J/\langle\kappa\rangle$ is the social interaction strength and $\langle\kappa\rangle$ the mean degree of the network of social interactions. The quantity β is the inverse of the social temperature that is a measure of the degree of confidence the members have in the other judgement/opinion. Similarly, the quantity β' is related to the level of confidence the members have about their perceived fitness (the higher β', the higher the confidence).

The pay-off function $\Delta V(\mathbf{s}'_l, \mathbf{s}_l)$ is simply the change of fitness perceived by the agent when its opinion on the decision j changes from s_l to $-s_l$. The group fitness value Eq. (1) is used as a measure of the performance of the collective-decision making process. To calculate the group fitness value, the vector $\mathbf{d} = (d_1, d_2, \ldots, d_N)$ needs to be determined. To this end, consider the set of opinions $\left(\sigma_1^j, \sigma_2^j, \ldots, \sigma_M^j\right)$ that the members of the group have about the decision j, at time t. The decision d_j is obtained by employing the majority rule, i.e. we set:

$$d_j = \mathrm{sgn}\left(M^{-1}\sum_k \sigma_k^j\right), \qquad j = 1, 2, \ldots, N \tag{5}$$

If M is even and in the case of a parity condition, d_j is, instead, uniformly chosen at random between the two possible values ± 1. The group fitness is then calculated as $V[\mathbf{d}(t)]$ and the ensemble average $\langle V(t)\rangle$ is then evaluated. The degree of consensus among the members is also computed. Following Ref. [53] this is defined as:

$$\chi(t) = \frac{1}{M^2 N}\sum_{j=1}^{N}\sum_{kh=1}^{M} R_{hk}^j(t) \tag{6}$$

where $R_{hk}^j(t) = \left\langle \sigma_k^j(t)\sigma_h^j(t)\right\rangle$. Observe that $0 \le \chi(t) \le 1$.

3. Criticality and swarm intelligence

The dynamics of the decision-making process is governed, for a group of fixed size M, by a tern of parameters, βJ, β' and p, such that a 3D phase diagrams would completely identify the behavior and performance of the group in making decisions. However, in order to simplify the discussion, we will fix the parameter p, since the qualitative behavior of what we describe does not change with this parameter, representing the steady-state response of the system, in terms of average fitness values $V_\infty = \langle V(t \to \infty)\rangle$ and degree of consensus $\chi_\infty = \langle \chi(t \to \infty)\rangle$, as a function of βJ and β'. Calculations have been carried out assuming that the network of social interactions on each decision layer is fully connected, i.e. $\langle \kappa \rangle = M - 1$. We simulate the Markov process by using the well-known stochastic simulation algorithm proposed by Gillespie [53,62,63]. For any given set of input parameters we computed 200 different realizations of the same process, calculating their ensemble average. An example is reported by the phase diagram shown in Fig. 1 for $N = 27,\ K = 23,\ M = 11,\ p = 1$. The emergence of the collective intelligence of the group, i.e. its ability to make decisions associated with high group fitness values, is identified by a $U-$ shaped critical transition front (dashed line in the Fig. 1), which provides, for each β' the corresponding critical value $(\beta J)_C$ of the social interaction strength that leads to a sudden and concurrent change from low to high or relatively high values of group fitness V_∞ and consensus χ_∞. On the critical front a minimum can be identified $\left\{\beta'_{\min}, (\beta J)_{\min}\right\}$, which represents the minimum value of reduced social strength βJ at which the collective intelligence of the group emerges, provided that the self-confidence level takes the right value $\beta' = \beta'_{\min}$. Very interesting is to observe that very high values of group fitness V_∞ can only be reached in a relatively small region [the red parabolic segment in Fig. 1(a)] near to the critical front and located close to the point $\left\{\beta'_{\min}, (\beta J)_{\min}\right\}$, belonging to the ordered region [see Fig. 1(b)] of the phase diagram where consensus among the individuals in the group is very high. This very specific region is the only part of the phase diagram where collective intelligence emerges. We refer to it as the collective intelligence region.

Interestingly, it can be shown [54] that the mutual information $MI(\chi_\infty, V_\infty)$ between the consensus χ_∞ and the fitness V_∞

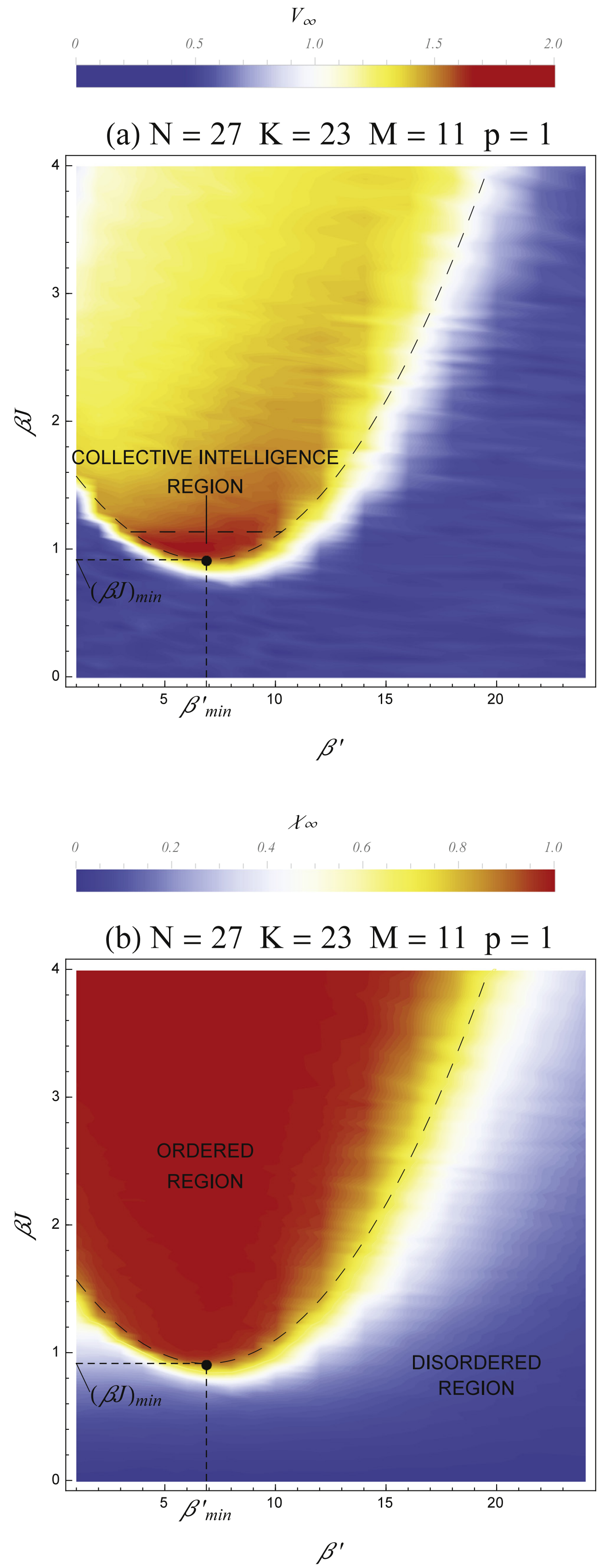

Fig. 1. The stationary values of the averaged fitness V_∞, (a); the averaged consensus χ_∞, (b) as a function of βJ and β'. Results are presented for $N = 27,\ K = 23,\ M = 11,\ p = 1$. The color bar identifies the values of V_∞ and χ_∞ respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

is strongly peaked just close to the collective intelligence region of the phase diagram. This evidently shows that, at criticality, the system is flexible, the agents explore well the fitness landscape, indirectly exchanging, through social interaction, a consistent amount of information about it. In this condition, the group, as a whole, acquires more knowledge about the fitness landscape compared to each single agent, and experiences the so-called collective intelligence state of the group [51].

4. The human group optimization algorithm

In this section we design the HGO algorithm exploiting the collective intelligence property of the decision making process to solve combinatorial problems. To this aim, we emulate the process followed to design the Simulated Annealing algorithm [16]. We first observe that the Markov process defined in Eq. (3) with transitions rates Eq. (4) converges to the stationary probability distribution [53]

$$P_0(\mathbf{s}_l) = \frac{\exp\left[-\beta E(\mathbf{s}_l) + 2\beta' V(\mathbf{s}_l)\right]}{\sum_k \exp\left[-\beta E(\mathbf{s}_k) + 2\beta' V(\mathbf{s}_k)\right]} \tag{7}$$

where the total level of conflict is $E(\mathbf{s}) = -\langle 2\kappa\rangle^{-1} J \sum_{ij} A_{ij} s_i s_j$, and $\bar{V}(\mathbf{s}_l) = V_k(\sigma_k)$ Eq. (7) is a Boltzmann distribution with effective energy

$$E_{\text{eff}}(\mathbf{s}_l) = -V(\mathbf{s}_l) + \alpha E(\mathbf{s}_l) \tag{8}$$

where $\alpha = \beta/(2\beta')$. Since we want to maximize the group fitness $V(\mathbf{s}_l) = V_k(\sigma_k^1, \sigma_k^2, \ldots, \sigma_k^N)$, where $s_l = \sigma_k^j$, $k = \text{quotient}(l-1, M) + 1$, and $j = \text{mod}\,(l-1, M) + 1$, we need to slowly increase the parameter β' during the process, in a similar fashion as in the case of Simulated Annealing (SA). We then make the parameter β' change during the process as follows

$$\beta' = \beta_0' \log(i+1) \tag{9}$$

where i is the time iterator and β_0' is set according to Ref. [64]. However, at the same time, we need to guarantee that the system undergoes a transition to the intelligence state and keep this state active, in order to exploit the advantage of the emergence of collective intelligence in finding the optimum of the group fitness. We obtain these by choosing

$$\beta J = \min\{\mu(i-1), (\beta J)_{\min}\} \tag{10}$$

where μ is chosen by the user. These conditions assure that the critical transition to the collective intelligence state is fully achieved during the optimization process, and that $\alpha = \beta/(2\beta')$ vanishes in the long term limit so as to allow $E_{\text{eff}}(\mathbf{s}_l) \to -\bar{V}(\mathbf{s}_l)$ as the optimization process advances. Note that, when the level of knowledge of individuals is $p = 1$, the two conditions Eqs. (9) and (10), akin the Simulated Annealing, make the proposed algorithm converge in probability to the optimum of $V(\mathbf{d})$ [65,66]. Also observe that by setting $\beta J = 0$, one obtains the classic multi-agent simulated annealing (MASA), which is characterized by the absence of social interactions among the agents, and, as such, unable to exploit the collective intelligence properties of the group.

5. Simulation and results

In this section we discuss the performance of the proposed optimization algorithm HGO in finding the global optimum of *NK* fitness landscapes. In all simulations, each stochastic process is simulated by generating 200 different realizations and the ensemble average of the results is calculated. The simulations are stopped at steady-state, i.e. when changes in the time-averages of consensus and pay-off over consecutive time intervals of a given length is sufficiently small. Calculations have been carried out for $N = 27$ and $K = 11, 17, 23$. The number of agents employed is $M = 11$. Different levels of knowledge in the range $0.1 \le p \le 1$ have been considered. The parameter $\beta_0' = T_0^{-1}$, i.e. the level of confidence of the agents about their perceived fitness at the beginning of the search process, has been computed following the algorithm proposed in Ref. [64]. We assumed, for the three landscapes considered, that an initial acceptance probability is $\psi_0 = 0.8$, thus obtaining the values $\beta_0' = 1.03,\ 0.83,\ 0.72$. Fig. 2 shows the evolution (i is the time iterator) in terms of fitness values $\langle V(i)\rangle$ and consensus $\chi(i)$ for the most complex case investigated, $N = 27$ $K = 23$. We notice that increase of fitness $\langle V(i)\rangle$ [Fig. 2(a)] is always accompanied by a simultaneous increase of the consensus $\chi(i)$ [Fig. 2(b)]. This confirms that, as required by the developed methodologies described in Section 4, during the optimization process, the critical transition to the collective intelligence state of the system always occurs. It is worth noting that increasing the level of knowledge p of the agents significantly enhances the performance of the optimization strategies in terms of V_∞, and also speeds up the convergence toward the steady state condition. Fig. 3 reports the steady-state values of the group fitness V_∞ [Fig. 3(a)], and consensus χ_∞ [Fig. 3(b)] of the HGO algorithm, as a function of the level of knowledge p, for $N = 27$, $K = 11, 17, 23$, and $M = 11$. We note that increasing the complexity of the fitness landscape determines only a slight deterioration of the performance of the HGO algorithm. Concerning the effect of p at given K, the trends in Fig. 3(a), show that increasing p has a beneficial effect in terms of optimization performance of the HGO algorithm. However as p is augmented the increase of HGO performance occurs at decreasing rate. Hence, for $p > 0$ a significant change in the performance of the algorithm can no longer be observed. This indicates that the enhancement of exchange of information among the agents about the group fitness landscape, which occurs at criticality (i.e. in the collective intelligence state), makes the system as a whole about the group fitness

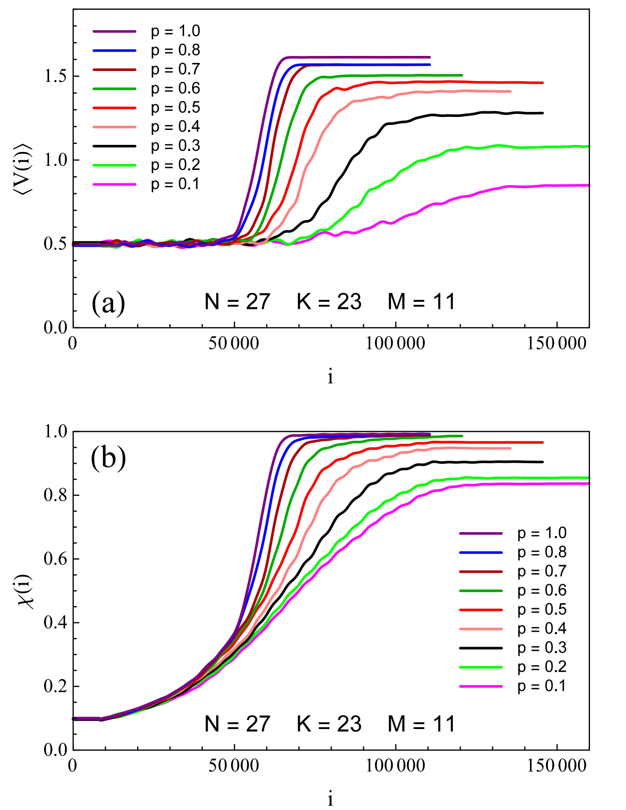

Fig. 2. The evolution of the average group fitness $\langle V(i)\rangle$, (a); and of the consensus $\chi(i)$, (b); for different levels of knowledge p, $M = 11$, $N = 27$ and $K = 11$.

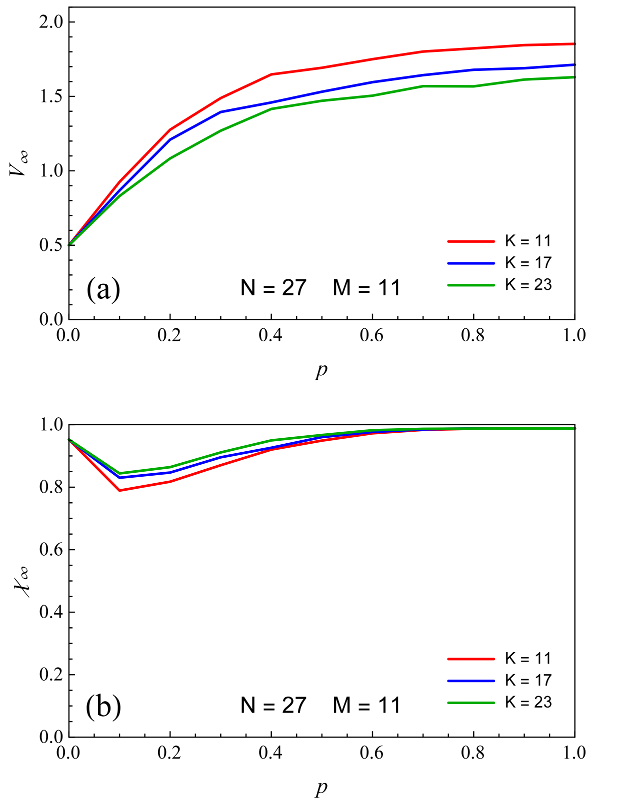

Fig. 3. The stationary values of the average group fitness V_∞, (a); and the consensus χ_∞, (b); as a function of p. Results are presented for $N = 27$, $M = 11$ and $K = 11, 17, 23$.

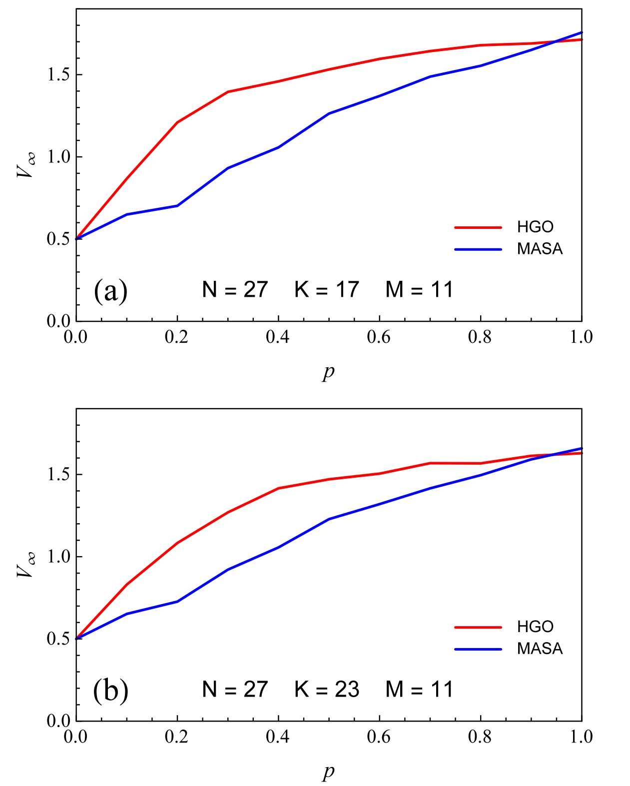

Fig. 4. A comparison between HGO and MASA, in terms of the fitness values V_∞ as a function of the knowledge level p, for $N = 27$, $K = 11, 17, 23$, and $N = 100$, $K = 15$.

even in the case of significantly low ($p \approx 0.5$) level of knowledge of the agents. This property is of utmost importance, as this allows this algorithm to strongly outperform other algorithms when the knowledge of the fitness group is only partial and each individual in the group only possess his own estimation of such fitness landscape (see Section 6).

Also we note [see Fig. 3(b)] that for vanishing values of p the consensus χ_∞ takes high values, because each agent's choice is driven only by consensus seeking. Increasing p determines an initial decrease of consensus, for the self-interest of each member leads to a certain level of disagreement. However, a further increment of p makes the members' knowledge partially 'overlap'. This pushes the agents, while being driven by self-interest, to make similar choices, and, therefore, to increase the level of consensus. The complexity parameter K slightly deteriorates the performance of the HGO optimization algorithm. In particular, for $p = 1.0$ and $N = 27$, values of $K = 11, 17, 23$ lead respectively to final fitness values $V_\infty = 1.85, 1.71, 1.63$. Recalling that the values of the *NK* fitness landscape are almost normally distributed with average $\bar{V} = 0.5$ and variance $\sigma_V^2 = 1/12$ (see Appendix A), it is worth noticing that the values $V_\infty = 1.85, 1.71, 1.63$ fall at a distance from the average value $\bar{V}$ of about four times the standard deviation σ_V of the fitness landscapes. This is a very remarkable results as the probability P of finding a fitness value larger than $V > \bar{V} + 4\sigma_V$ is really tiny, i.e. $P\left(V > \bar{V} + 4\sigma_V\right) = 3.17 \times 10^{-5}$.

6. Comparison with other optimization algorithms

In this section we compare the performance of the proposed HGO algorithm with the multi agent simulated annealing (MASA) algorithm. We recall that the latter is characterized by the absence of social interactions among the agents, therefore in this case the choice of the decision vector is made by comparing the fitness values associated with the agents and by choosing the solution which provides the better fitness. On the other hand, in the HGO decisions are chosen by enforcing the majority rule. This is not an irrelevant difference, as in choosing the solution HGO never has the need to compare the fitness values associated with each agents. This makes the algorithm work also when the agents have only a guess about the group fitness landscape (partial knowledge). In our opinion, this is one of the most striking and novel features of the proposed optimization algorithm. In Fig. 4 we show the results of the optimization for $N = 27$ and $K = 17, 23$, with p ranging from 0 to 1. A group size of $M = 11$ is adopted in all cases. In all simulations, each stochastic process is simulated by generating 200 different realizations and by taking the ensemble average of the results. We note that HGO outperforms MASA on the entire range of level of knowledge p, except at $p \approx 1$. In fact, in the case of limited knowledge, the social interactions push those agents in the group with no knowledge about a certain decision, to seek consensus and make the choices of individuals, who know the influence of the given decision on the fitness. This clearly explains the benefit of properly including social interactions into the optimization algorithm, and also clarifies why the entire group perform much better compared to the case of non-socially interacting members (MASA algorithm). Of course for $p \approx 1$ the presence of social interaction may be slightly detrimental, as in making the final decision, employing the majority rule may be less effective than comparing directly the fitness of each agents (which at $p \approx 1$ coincides with the fitness of the entire group) and choosing the best values. We stress that the considerable gap between the HGO and MASA algorithm, in the case of limited cognitive level of the individuals in the group, is, as a matter of fact, a consequence of the emergence of collective intelligence, which the MASA cannot profit from.

We also tested the efficacy of HGO algorithm with the classic simulated annealing (SA) and genetic algorithm (GA), both particularly suited to solving combinatorial discrete optimization prob-

Table 1
Comparison of the final fitness values V_∞, found by HGO, MASA, SA and GA, for the cases $N = 27$, $K = 11, 17, 23$, and $N = 100$, $K = 15$.

$p = 1$	HGO	MASA	SA	GA
$N = 27$ $K = 11$	1.853	1.888	1.473	1.565
$N = 27$ $K = 17$	1.713	1.757	1.239	1.456
$N = 27$ $K = 23$	1.629	1.658	1.153	1.381
$N = 100$ $K = 15$	2.660	2.799	1.646	2.420

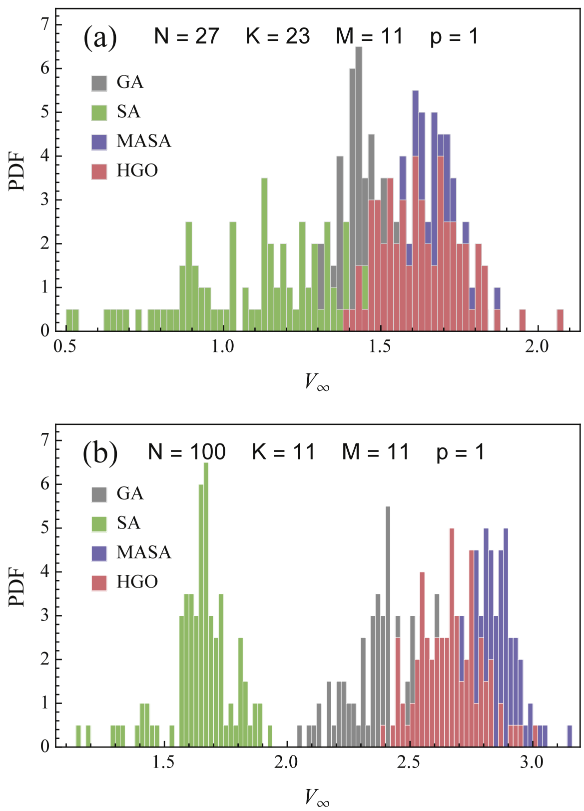

Fig. 5. The distribution of the fitness values V_∞ found by HGO, MASA, SA and GA respectively. Results are showed for $N = 27$, $K = 17$ and $N = 100$, $K = 15$.

lems. In the case of GA, we utilized the Global Optimization Toolbox of MATLAB® R2015b with 100 agents (recall that for the other methods we just used 11 agents). The set of parameter utilized in the GA algorithm are fully reported in Appendix B. Simulations have been carried for $p = 1$. Table 1 shows a comparison of the four optimization methods in terms of the final fitness values V_∞ found during the simulations, for $N = 27$, $K = 11, 17, 23$, and $N = 100$, $K = 15$. We note that the HGO algorithm always outperforms the other algorithms, except (recall that $p = 1$) the MASA algorithm. Indeed, the latter, as already mentioned so far, presents slightly better performance than the HGO at knowledge level $p \approx 1$. Fig. 5 show the histograms of the probability density function of the final fitness values V_∞ found during the 200 replications, by the four mentioned algorithms, for the cases of $N = 27$, $K = 23$, and $N = 100$, $K = 15$. As already noted, at $p = 1$, only the MASA algorithm slightly outperforms the HGO method here proposed.

7. Conclusions

In this paper we proposed a novel collective intelligent-based optimization algorithm, mimicking the collective decision-making of human groups. This algorithm, which we termed Human Group Optimization (HGO) method, describes the decision process of the agents in terms of a time-continuous Markov chain, where the transition rates are defined so as to capture the effect of the self-interest, which pushes each single agent to increase his/her perceived fitness, and of social interactions, which stimulate member to seek consensus with the other members of the group. The Markov chain is, then, characterized by a couple of parameters that, likewise the Simulated Annealing, are subjected to a specific cooling schedule that in the long-term limit makes the system converge in probability to the optimal value. The choice of the parameters is made in order to guarantee the transition to a consensus state at which the group of agents shows a very high degree of collective intelligence. While being in this state, the agents explore the landscape by sharing information and knowledge through social interactions, so as to achieve very good solutions even in the case of limited knowledge.

To test the proposed HGO algorithm, we considered the hard-*NP* problem of finding the optimum on *NK* fitness landscape and compared the methodology with other well established algorithms as the Genetic Algorithm (GA), the Simulated Annealing (SA) and the Multi-Agent Simulated Annealing (MASA). In all cases the HGO has been shown to significantly outperform the other algorithms over the entire range of cognitive level of the agents. For $p \approx 1$, as expected, only MASA slightly outperforms the HGO.

Summarizing, our algorithm presents several advantages that make it very suitable to solve complex problems. It is flexible because it can be used to solve almost any combinatorial problem. However, its most attractive feature relies in its ability to identify very good solutions in presence of limited or partial knowledge of the fitness landscape. For this reason it appears very promising for applications in distributed decision making contexts. Furthermore, while the vast majority of collective intelligent algorithms, mimicking the behavior of social groups like insects and animals, are based on the mechanism of the stigmergy, our algorithm introduces a mechanism based on the direct communication among individuals, which is a more powerful and effective way to achieve coordination. Under this perspective, the proposed code is novel and unique within the class of collective intelligent optimization codes.

We recognize that this first version of the algorithm could be further improved in future research by identifying better cooling schedules. The algorithm could be also fine-tuned to solve specific operations management problems characterized by distributed decision making and information asymmetry, such as multi-stage production scheduling, location routing problem, supply chain inventory problem, just to name a few. Additional numerical tests and theoretical investigation, not in the scope of present study, are however needed to quantify pros and cons.

Acknowledgments

The authors warmly thanks ARO for support through Grant No. W911NF-15-1-0245.

Appendix A. The fitness landscape

In the *NK* model a real valued fitness is assigned to each bit string $\mathbf{d} = (d_1, d_2, \ldots, d_N)$, where $d_i = \pm 1$. This is done by first assigning a real valued contribution W_i to the *i*th bit d_i, and then by defining the fitness function as $V(\mathbf{d}) = N^{-1} \sum_{i=1}^{N} W_i\left(d_i, d_1^i, d_2^i, \ldots, d_K^i\right)$. Each contribution W_i depends not just on i and d_i but also on K $(0 \leq K < N - 1)$ other bits. Now let us define the substring $\mathbf{s}_i = \left(d_i, d_1^i, d_2^i, \ldots, d_K^i\right)$, by choosing at random, for each bit *i*, K other bits. The number of contributions $W_i(\mathbf{s}_i)$ is equal to the number of different values that can be enumerated with the substring of $k + 1$ binary elements, i.e. it is 2^{K+1}.

Table 2
The options set, different from the default values, imposed on the Global Optimization Toolbox of MATLAB^R R2015b to test the Genetic Algorithm (GA) in solving the proposed *NK* optimization problems, $N = 27,\ K = 11, 17, 23,$ and $N = 100,\ K = 15$.

	Type	Bit string
	Size	100
Selection	Roulette	
Mutation	Gaussian	
Crossover	Function	Two points
Migration	Direction	Both
	Generations	10^5
Stopping	Stall generations	5000
Criteria	Stall Test	Geometric weighted
	Function tolerance	10^{-8}

Each single value $W_i(\mathbf{s}_i)$ is its value is drawn from a uniform distribution, usually in the range [0, 1]. Thus, a random table of $N \times 2^{K+1}$contributions is generated independently for each *i*th bit, allowing the calculation of the fitness function $V(\mathbf{d})$. The reader is referred to Refs. [56,57] for more details on the *NK* complex landscapes. Notice that increasing the complexity $C = K + 1 + \log_2 N$, not only affects the number of local maxima, but also the autocorrelation of the landscape itself. In particular at the maximum level of complexity i.e. when $K = N - 1$, one can show that the number of local maxima is $2^N/(N+1)$ and that the fitness values are completely uncorrelated with each other, in this case the fitness landscape is represented by a isotropic white noise. This means that using *NK* model it is not possible to control separately the complexity, the autocorrelation and the actual level of anisotropy of the landscape. Also, it is worth noticing that the stochastic fitness function $V(\mathbf{d})$, being the mean value of several independent uniformly distributed contributions of expectation value $\bar{W} = \langle W \rangle$ and variance $\sigma_W^2 = \left\langle \left(W - \bar{W}\right)^2 \right\rangle$, is very well approximated, as prescribed by the central limit theorem, by a Gaussian distribution with average $\langle V \rangle = \bar{W}$ and variance $\sigma_V^2 = \left\langle \left(V - \bar{W}\right)^2 \right\rangle = \sigma_W^2/N$. Thus, increasing the number of decisions *N* leads to a decrease of σ_V^2, so that for very large *N* the distribution of fitness values *V* degenerates into a Dirac delta distribution centered in $\bar{W}$. To prevent this from occurring we preferred to rescale the fitness values *V* in such a way to keep the same average $\bar{W}$ and the same variance σ_W^2, i.e. we use

$$V \rightarrow \bar{W} + \sqrt{N}\left(V - \bar{W}\right) \tag{A1}$$

Appendix B. Genetic algorithm options

The options we used to perform GA calculations within the Global Optimization Toolbox of MATLAB^R *R*2015*b* are reported in Table 2.

For the meaning of each parameter in the table, the reader should refer to the MATLAB guide [67].

References

[1] Lee CH, Lucas A. Phys Rev E 2014;90:052804.
[2] Brummitt CD, Chatterjee S, Dey PS, Sivakoff D. Ann Appl Probab 2015;25(4):2013–38. doi:10.1214/14-AAP1041.
[3] Galam S. Physca A 1997;238:66–80.
[4] Galam S. Martins. a. c. r.. Phys Rev E 91 2015:012108.
[5] Koraimann G, Wagner MA. Front Cell Infect Microbiol4:54 2014. doi:10.3389/fcimb.2014.00054.
[6] Popat R, Crusz SA, Stephen P. Diggle 2008;87(1):63–75. doi:10.1093/bmb/ldn030.
[7] Westerhoff HV, Brooks AN, Simeonidis E, et al. Front Microbiol 2014;5(379). doi:10.3389/fmicb.2014.00379.
[8] Krieger MJB, Billeter J-B, Keller L, Nature 2000 406, 992–995, doi:10.1038/35023164.
[9] Rubenstein M, Cornejo A, Nagpal R. Science 2014;345(6198):795–9. doi:10.1126/science.1254295.
[10] Werfel J, Petersen K, Nagpal R. Science 2014;343(6172):754–8. doi:10.1126/science.1245842.
[11] Brambilla M, Ferrante E, Birattari M, Dorigo M. Swarm Intell 2013;7:1–41. doi:10.1007/s11721-012-0075-2.
[12] Bonabeau E, Dorigo M, Theraulaz G. Nature 2000;406:39–42. doi:10.1038/35017500.
[13] Garnier S, Gautrais J, Theraulaz G. Swarm Intell 2007;1:3–31. doi:10.1007/s11721-007-0004-y.
[14] Vanni F, Luković M, Grigolini P. Phys Rev Lett 2011;107:078103. doi:10.1103/PhysRevLett.107.078103.
[15] Easley D., Kleinberg J.. Cambridge university press. 2010.
[16] Kirkpatrick S, Gelatt JrCD, Vecchi MP. Optimization by simulated annealing. Science 1983;220(4598):671–80. doi:10.1126/science.220.4598.671.
[17] Dorigo M. Optimization, learning and natural algorithms. Politecnico di Milano, Milan; 1992. Ph.d. thesis. Available: http://www.ci.nii.ac.jp/naid/10016599043/.
[18] Dorigo M, Maniezzo V, Colorni A. The ant system:optimization by a colony of cooperating agents. IEEE Trans Syst, Man, Cybern–Part B 1996;26(1):1–13.
[19] Dorigo M, Gambardella LM. Ant colonies for the traveling salesman problem. BioSystems 1997;43:73–81.
[20] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks. IV; 1995. p. 1942–8.
[21] Storn R, Price K. Differential evolution, a simple and efficient heuristic for global optimization over, continuous spaces. J Global Optim 1997;11(4):341–59. doi:10.1023/A:1008202821328. 341.
[22] Karaboga D. An idea based on honeybee swarm for numerica optimization. Technical Report TR06. Erciyes University; 2005.
[23] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Global Optim 2007;39:459–71.
[24] Krihnanand KN, Ghose D. Glowworm swarm optimization for simultaneous capture of multiple local, optima of multimodal functions. J Swarm Intell 2009;3(2):87–124.
[25] Krihnanand KN, Ghose D. Glowworm swarm optimization: a new method for optimizing multi-modal function. J Comput Intell Stud 2009;1(1):93–119. doi:10.1504/IJCIStudies.2009.02534.
[26] Yang X-S, Deb S. Engineering optimisation by cuckoo search. Int J Math Modell Numer Optim 2010;1(4). doi:10.1504/IJMMNO.2010.035430.
[27] Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Software 2014;69:46–61. doi:10.1016/j.advengsoft.2013.12.007.
[28] Mirjalili S. The ant lion optimizer. Adv Eng Software 2015;83:80–98. 5//.
[29] Grigolini P. Call for papers: special issue on evolutionary game theory of small groups and their larger societies. Chaos, Solitons Fractals 2017;103:371–3. doi:10.1016/j.chaos.2017.06.020.
[30] AbWahab MN, Nefti-Meziani S, & Atyabi A. A comprehensive review of swarm optimization algorithms. PLoS ONE 2015;10(5):e0122827. doi:10.1371/journal.pone.0122827.
[31] Bianchi L, Dorigo M, Gambardella LM, Gutjahr WJ. A survey on metaheuristics for stochastic combinatorial optimization. Nat Comput 2009;8(2):239–87. doi:10.1007/s11047-008-9098-4.
[32] Zhang R, Song S, Wu C. A hybrid artificial bee colony algorithm for the job shop scheduling problem. Int J Prod Econ 2013;141(1):167–78. doi:10.1016/j.ijpe.2012.03.035.
[33] Hamta N, Ghomi SMTF, Jolai F, Shirazi MA. A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect. Int J Prod Econ 2013;141(1):99–111. doi:10.1016/j.ijpe.2012.03.013.
[34] Ottino JM. Engineering complex systems. Nature 2004;427:399. doi:10.1038/427399a.
[35] Bonabeau R, Dorigo M, Theraulaz G. Swarm intelligence: from natural to artificial systems. New York, NY: Oxford University Press, Santa Fe Institute Studies in the Sciences of Complexity; 1999. Paper: ISBN 0-19-513159-2.
[36] Dorigo M, Di Caro G, Gambardella LM. Ant algorithms for discrete optimization. Artif Life 1999;5(2):137–72.
[37] Ferretti I, Zanoni S, Zavanella L. Production–inventory scheduling using ant system metaheuristic. Int J Prod Econ 2006;104(2):317–26. doi:10.1016/j.ijpe.2005.01.008.
[38] Nia AR, Far MH, Niaki STA. A fuzzy vendor managed inventory of multi-item economic order quantity model under shortage: an ant colony optimization algorithm. Int J Prod Econ 2014;155:259–71. doi:10.1016/j.ijpe.2013.07.017.
[39] Moncayo-Martinez LA, Zhang DZ. Multi-objective ant colony optimisation: a meta-heuristic approach to supply chain design. Int J Prod Econ 2011;131(1):407–20. doi:10.1016/j.ijpe.2010.11.026.
[40] RJG C, Krause S, von Engelhardt N, Fariann JJ, Krause J, RHJM K. Collective cognition in humans: groups outperform their best members in a sentence reconstruction task. PLoS ONE 2013;8(10):e77943. doi:10.1371/journal.pone.0077943.
[41] Couzin ID. Collective cognition in animal groups. Trends Cogn Sci 2009;13(1):36–43. doi:10.1016/j.tics.2008.10.002.
[42] Sumpter DJT, Pratt SC. Quorum responses and consensus decision making. Phil Trans R Soc B 2009;364:743–53. doi:10.1098/rstb.2008.0204.
[43] Ward AJW, Sumpter DJT, Couzin ID, Hart PJB, Krause J. Quorum decision-making facilitates information transfer in fish shoals. PNAS 2008;105(19):6948–53. doi:10.1073/pnas.0710344105.

[44] Arganda S, Pérez-Escudero A, de Polavieja GG. A common rule for decision making in animal collectives across species. PNAS 2012;109(50):20508–13.
[45] Ward AJW, Herbert-Read JE, Sumpter DJT, Krause J. Fast and accurate decisions through collective vigilance in fish shoals. PNAS 2011;108(6):2312–15. doi:10.1073/pnas.1007102108.
[46] Pérez-Escudero A, de Polavieja GG. Collective animal behavior from bayesian estimation and probability matching. PLoS Comput Biol 2011;7(11):e1002282. doi:10.1371/journal.pcbi.1002282.
[47] Watts DJ. A simple model of global cascades on random networks. PNAS 2002;99(9):5766–71. doi:10.1073/pnas.082090499.
[48] Turalska M, Lukovic M, West BJ, Grigolini P. Complexity and synchronization. Phys Rev E 2009;80:021110. doi:10.1103/PhysRevE.80.021110.
[49] Wang Z, Szolnoki A, Perc M. Optimal interdependence between networks for the evolution of cooperation. Sci Rep 2013;3(2470). doi:10.1038/srep02470.
[50] Wang Z, Szolnoki A, Perc M. Interdependent network reciprocity in evolutionary games. Sci Rep 2013;3:1183. doi:10.1038/srep01183.
[51] Woolley, Williams A, Chabris CF, Pentland A, Hashmi N, Malone TW. Evidence for a collective intelligence factor in the performance of human groups. Science 2010;330(6004):686–8.
[52] Engel D, Woolley AW, Jing LX, Chabris CF, Malone TW. Reading the mind in the eyes or reading between the lines? Theory of mind predicts collective intelligence equally well online and face-to-face. PLoS ONE 2014;9(12):e115212. doi:10.1371/journal.pone.0115212.
[53] Carbone G, Giannoccaro I. Model of human collective decision-making in complex environments. Eur Phys J B 2015;88(12):339–48. doi:10.1140/epjb/e2015-60609-0.
[54] De Vincenzo I, Giannoccaro I, Carbone G, Grigolini P. Criticality triggers the emergence of collective intelligence in groups. Phys Rev E 2017;96(2):022309.
[55] DiMaggio PJ, Powell WW. The iron cage revisited: institutional isomorphism and collective rationality in organizational fields. Am Sociol Rev 1983;48(2):147–60.
[56] Kauffman S, Levin S. Towards a general theory of adaptive walks on rugged landscapes. J Theor Biol 1987;128(1):11–45. doi:10.1016/S0022-5193(87)80029-2.
[57] Kauffman S, Weinberger E. The NK model of rugged fitness landscapes and its application to the maturation of the immune response. J Theor Biol 1989;141(2):211–45. doi:10.1016/s0022-5193(89)80019-0.
[58] Weinberger ED. NP Completeness of kauffman's n-k model, a tuneably rugged fitness landscape; 1996. Santa Fe Institute working paper
[59] Glauber RJ. Time-dependent statistics of the ising model. J Math Phys 1963;4:294–307. doi:10.1063/1.1703954.
[60] Weidlich W. Physics and social science - the approach of synergetics. Phys Rep 1991;204(1):1–163.
[61] Schweitzer F. Brownian agents and active particles. Collective dynamics in the natural and social sciences. Berlin-Heidelberg-New York: Springer Series in Synergetics - Springer Complexity, Springer; 2007. Isbn: 978–3540738442
[62] Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J Comput Phys 1976;22(4):403–34. doi:10.1016/0021-9991(76)90041-3.
[63] Gillespie DT. Exact stochastic simulation of coupled chemical reactions. J Phys Chem 1977;81(25):2340–61. doi:10.1021/j100540a008.
[64] Ben-Ameur W. Computing the initial temperature of simulated annealing. Comput Optim Appl 2004;29(3):369–85. doi:10.1023/B:COAP.0000044187.23143.bd.
[65] Geman S, Geman D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 1984(6):721–41. doi:10.1109/TPAMI.1984.4767596. PAMI-6.
[66] Hajek B. Cooling schedules for optimal annealing. Math Oper Res 1988;13(2):311–29.
[67] 2015. Global Optimization Toolbox User's Guide, MATLAB®R2015b, The MathWorks, Inc. 3 Apple Hill Drive, Natick, MA 01760-2098, Revised for Version 3.3.2 (Release 2015b).