Algorithms Inspired in Social Phenomena

Antonio Neme and Sergio Hernández

Abstract. Natural computing finds its source of inspiration in diverse biological phenomena and social behaviors from mainly insects and birds. In this chapter, we instead propose human social phenomena. The presented algorithms have been applied in optimization endeavours with success or are promising tools in the design of optimization techniques.

1 Introduction

Computer Science has turned its attention to a wide variety of natural phenomena with the aim of abstracting new optimization algorithms. Such phenomena can either be physical, biological or even social processes, like simulated annealing, genetic algorithms and interactions between large communities of insects respectively.

Social phenomena have been extensively studied as a basis for several algorithms, many of them in the area of optimization. The great majority of these social phenomena come from the social behavior shown by ants, termites and some other insects, as well as from bird flocks and fish schools [9]. Most of the social phenomena that have been used as inspiration for optimization algorithms are mainly cases of collaboration between organisms, and are referred to as swarm intelligence [10].

The relevance, complexity and computational power of optimization algorithms taken from social phenomena in species other than humans (swarm intelligence) is enormous and can be traced elsewhere, as, for example, in [15, 19, 64]. In this

Antonio Neme
Non-lineal Dynamics and Complex Systems Group,
Universidad Autónoma de la Ciudad de México
e-mail: antonio.neme@uacm.edu.mx

Sergio Hernández
Posgraduate Program in Non-linear Dynamics and Complex Systems,
Universidad Autónoma de la Ciudad de México
e-mail: cheko@ciencias.unam.mx

R. Chiong (Ed.): Nature-Inspired Algorithms for Optimisation, SCI 193, pp. 369–387.
springerlink.com

chapter we discuss social phenomena in human societies that have been or may be the foundation for optimization algorithms.

2 Social Phenomena

Since their early stages, human societies have proven to be very successful in several, if not all, aspects of technological development [6]. People have been optimizing every aspect of daily life, from water supply for agriculture to machine design and transportation, to name just a few. Many of these processes are the result of a group of people explicitly working on them. In this sense, a group of say, engineers working toward optimizing a mechanical engine is a social phenomenon. On the other hand, some social phenomena not explicitly intended to reach a certain goal could also be the source of inspiration for optimizing algorithms. We are interested in those social phenomena that are the result of a non-conscious procedure, that is, social phenomena in which there is not an explicit agreement between individuals to perform a given optimizing algorithm.

The study of social phenomena has been dictated by the cultural influence of their time. Human behavior and social phenomena were analyzed at the light of pendulums, steam engines, computers and so on [6]. The complex systems theory establishes that the behavior of a system is not accessible from the study of its separated components [11]. To fully understand the richness of complex systems, it is mandatory to explicitly define the relations that are observed between the components. Almost all these relations are non-linear and in many cases conflicts among components are present.

Modern social sciences have recently incorporated the multiagent-based modeling, which has shed new light on some unclear aspects in social behavior [11]. In multiagent models, actors of the studied process are abstracted into a group of agents whose behavior is guided by a set of rules that represents the relationship observed among the actors. There is a two-way communication bridge between social and computer sciences, as social phenomena have been the basis for several algorithms [15, 19].

The main idea behind Algorithms Inspired in Social Phenomena is that the computational power of the inspired algorithms is correlated to the richness and complexity of the social behavior. Social phenomena arise as the result of interaction among individuals. These interactions may be non-linear and the number of interacting agents is required to surpass a given threshold. In general, the agents may not be aware of the state of all other agents.

By optimization we refer to finding a minimum or a maximum of a given function. It is defined over a triplet (S, Ω, f) in which S is the search space defined by a set of variables, Ω is the set of constraints over the variables and f is the objective function to be optimized.

The objective function to minimize (maximize) is

$$f(x) \tag{1}$$

subject to constraints $\Omega = g_i \cup h_j$, of which there are k inequalities:

$$g_i(x) \geq 0, i = 1, ..., k \tag{2}$$

and r equalities:

$$h_j(x) = 0, j = 1, ..., r \tag{3}$$

In this chapter we present social phenomena that are present mainly in human societies and that have motivated several optimization algorithms. We also refer to some social processes whose metaphor may lead to new algorithms. The hypothesis is that some of these phenomena, the ones with high complexity, have more computational power than other, less complex phenomena.

Human interactions cannot be explained only by means of biological information, as a great variety of social phenomena are present in societies [35]. Human social phenomena tend to show a high level of diversity. Even if there is not an optimizing principle behind social interactions they lead to robust social structures that may even present a high level of stability. The former features make social phenomena a valuable source of inspiration for algorithms.

In the following sections, the human social phenomena we present are leadership and influence from prominent counterparts, alliance formation, neighborhood segregation, and social labeling of individuals.

3 Leadership

Leaders have been important for societies since the dawn of humankind. In the early ages, wizards and magicians, followed by feudal lords and religious officers, and now presidents or football stars, determine many of the individual behaviors and, in some sense, are the idols to whom many people are attracted [34]. Leaders have been identified as outstanding members of a society and they tend to do better than the rest. Leaders influence their counterparts who are drawn toward the former's position, circumstance that may be seen as a searching strategy.

3.1 Society Civilization Algorithm

Several algorithms have been inspired by the leadership phenomena. For example in [43], an optimization algorithm is defined in terms of leader guidance, the so called *society civilization algorithm* (SCA). In this model search space is explored by candidate solutions which at the beginning are distributed in clusters that resemble societies. The best solutions in each society bias the search toward them, that is, the leaders influence their counterparts to follow them. Some leaders may migrate to other regions and thus, their counterparts in former societies migrate with them. The set of all societies is defined as the civilization, in which all individuals (solutions) may interact by means of their leaders. The general algorithm of these schemes is summarized as:

1. t ← 0.

2. Generate the civilization $C(t)$ of N individuals: $C(t) = I_1, ..., I_N$ uniformly distributed in the parameter space.

3. Evaluate individuals by computing the objective function as well as constraints.

4. Construct $S(t)$ societies from $C(t)$ as clusters. Clusters may be formed by any cluster analysis technique.

5. Identify leaders in each society.

6. Migrate individuals in each society toward the location of its nearest leaders.

7. Identify leaders in the civilization $C(t)$ from the leaders of the societies $S(t)$.

8. Migrate society leaders toward the location of civilization leaders.

9. t ← t + 1.

10. If the stop condition is not met, go to step 3.

The social phenomena that inspired SCA are i) migration and ii) leadership, but also iii) cooperation: leaders share their knowledge, that is their position in search space, to the rest of individuals. From these features, SCA algorithms have been applied in several areas. For example, in [47], a variant of SCA is applied to optimize the economic dispatch with multiple minima, a well-known problem in electric power systems operation, and results are promising as performance is comparable to those from mathematical programming, with less computational effort. In [58], a variation of the particle swarm optimization model that incorporates the concept of leadership by allowing particles to move toward the location of the best evaluated particle lead to good results in several benchmarks, while reducing the number of collisions between particles.

In the context of data analysis, several techniques may be classified as leader-guided, as that of self-organizing maps [31], in which the so-called best matching unit attracts toward them the weight vectors of their neighbor units. An explicit leader-guided algorithm has been proposed in [56]. The proposed algorithm obtains a hierarchical structure from data in which each leader is the centroid of a cluster and there are one or more subleaders within that cluster.

3.2 Cultural Algorithms

A unique aspect that characterizes all human societies is the concept of culture. Culture is based on learning from experienced individuals and in that sense, there is also a guidance (leadership) from them to the rest of the individuals. Reynolds [44] introduced a type of algorithm based on how cultures gather information to solve the problems presented to them; these are called cultural algorithms (CA). Such algorithms are a vehicle for modeling social evolution and learning. The intention is to abstract the necessary knowledge from experiences needed to solve some specific problem into, as Reynolds calls it, a space belief which can vary among normative, spatial, temporal domain and exemplar knowledge.

The main idea behind CA is to divide the process of learning and information retrieving into three phases. First, a coarse-grained phase is established which is expected to grasp a general idea of the problem in order to identify regions to explore. Then, a fine grained phase and finally a phase that comes into action when the search process gets stagnated.

The approach to problem optimization inspired by cultural algorithms is an abstraction of how cultures learn to solve their problems by means of a space belief and the proper use of this space belief in order to retrieve useful facts related to the problem at hand. The intention is to abstract a methodology used by humans to optimize their solutions and this can only be done when the variables and the problem domain are fully understood.

Cultural algorithms are inspired by human behavior. They were proposed to incorporate some statements about cultural change. These determine a belief space, which, together with the population space, constrains the search space. The cultural algorithm is also guided by an influence scheme that prevents some individuals from modifying the belief space, while allowing others to do so. In this sense there is a desired, although unknown, behavior that is represented as a location that should be reached by individuals.

Cultural algorithms have been applied in multiobjective optimization, as for example, that of [16, 55], in which a CA is combined with evolutionary programming (CAEP) that achieves outstanding performance. The CAEP is sketched as:

1. Generate k individuals that are the initial population.
2. Evaluate the initial population.
3. Initialize the belief space.
4. While stop conditions are not met:
5. Apply mutation to generate p offspring.
6. Evaluate each offspring.

7. Obtain the relative performance of each solution by means of random mutations.

8. Select the q individuals with the largest number of victories to produce the new generation.

9. Add the non-dominated individuals to an external memory.

10. Modify the belief space with individuals in external memory.

It is important to realize that the cited algorithms seek to improve the performance of individuals in optimization problems. This is achieved by means of learning from the ”good” individuals. These algorithms complement some other convergence-oriented models that try to avoid unfeasible regions of the search space. In that sense, there is also a guide from the good individuals or leaders to the rest of the population.

In leader-based algorithms, such as SCA and CA, an important assumption is made about individuals: their inability to consider self experiences, that is, they are guided toward the leaders' location without relying on their own previous experiences (locations). Apart from leadership, migration toward leaders and cooperation from leaders to the rest of individuals, free-will has also been included in some algorithms, as that in [17], in which there is also a liberty parameter that allows or stops the individuals migrating to the region occupied by the leader. From this liberty of choosing the desired region to move to, a different set of algorithms have been proposed: the leaderless algorithms.

3.3 Leaderless Algorithms

In Leaderless Algorithms (LA) there are not privileged individuals that bias the direction of searching nor attract others toward them. In LA all individuals have a lack of global information and tend to present the same capabilities while communication is a necessary condition for the algorithm to be successful. In [18], a parallel searching algorithm is presented based on homogeneous agents that are able to construct a map of undirected graphs without a supervisor. The algorithm is a distributed version of the standard depth-first algorithm, but as there are several anonymous agents, they are not able to distinguish between nodes visited by themselves and nodes visited by others. Each agent generates a partial map, T_i, and the complete map of the graph is obtained by a spanning algorithm executed by the first agent that finishes its exploration.

In [48], a group of moving agents reaches a stable velocity as a consensus among all agents. No agent's preferences are identified as the desired option and in a scenario of unbounded time, consensus is always reached. Agent's individual performance cost is minimized and constrained to partial information and by doing so, the overall cost of reaching a stable and safe velocity is minimized.

LA has a strong inspiration in the family of algorithms known as swarm intelligence [10], but also in some human behaviors such as those observed in certain business and corporative firms in which duties and privileges are equally shared, as described in [40].

Leaders tend to group individuals toward them. Grouping is also a complex behavior in human societies and has been a source of inspiration for several algorithms. A leader may form a group, but leadership is not the only factor that causes group formation, as will be shown in the next section.

4 Alliance Formation Basis

An alliance, also known as a coalition, is relatively stable group of agents. Agents may be people, political parties, software programs, robots or firms that work toward a common goal that in general cannot be achieved by the agents on their own.

Humans and groups of humans tend to form alliances in several contexts. For example, in democratic regimes with multi-parties such as in Europe and Latin America, alliance formation is a very common practice. An alliance, or coalition, is a set of agents (humans or groups) that may have a similar set of features and a common goal, that in general contrast with goals and features of a rival alliance [21]. Agents may seek their own good and form alliance only by their selfish interests but alliance formation is only profitable when all agents gain more by being part of the group than interacting on their own. An agent might join one alliance not only because it is wealthy but also because by making the alliance wealthier, it increases the chances to defeat the other one in which rival agents may be clustered.

4.1 Basic Definitions

Coalition formation has been studied extensively in game theory mainly with the unfeasible condition that all participants possess full information about the features and preferences of each agent. From the multi-agent community, several proposals have been made with more feasible constraints, such as that of a lack of complete information about the rest of participants, although not all algorithms guarantee stable coalition formation. In [3], an algorithm that leads to stable alliances even with incomplete information is proposed, and since then, several algorithms with more constraints have been studied.

An alliance is goal-directed and in general short-lived [27]. Human alliances may also be very stable at short times, but very unstable after a critical time which translates into rifts and reconfigurations [22]. Alliance or coalition formation has been extensively studied in Computer Science, as there are many applications in which several agents work toward a common goal [42].

In general, the problem of alliance formation can be stated as finding the best partition such that all agents belong to one and only one coalition and at the same time a set of constraints is satisfied. The constraints are derived from the preferences of agents to interact with specific agents, or with agents that present specific features, while avoiding being in the same partition with agents that show some non-desirable features. Agents have features that describe them and allow the interaction with others. In the political scene, an agent, i.e. a party, may have a certain position about public education, public welfare, and science and research policies that determine those parties to whom it may ally with.

In the traditional alliance formation problem, there are only two alliances, which lead to a total of 2 possible alliance ensembles. Each point in the alliance space has a given energy, or feasibility. Some ensembles may not be very probable in the sense that some of their members may have very different characteristics. For example, a possible party alliance may include the left-most and the right-most parties, but as they show very different positions on the same issues, this alliance is very energetic, i.e., not feasible.

The space of possible ensembles was studied by Axelrod and Bennett under the name of landscape theory [4]. The landscape defined by all the possible ensembles may have several local minima according to the energy, but also may have a global minimum and thus, the ensemble be optimal. The quantity that is optimized here is the energy. Fig. 1 shows a landscape for a fictitious group of 9 political parties and their positions about certain social aspects. There are 256 possible alliances and in fig. 1-b, the energy for each one of them is shown.

Party	Public education	Social welfare	Abort	Religious liberty	Public companies	Intern. Affairs	Laboral issues	Oil issues
P1	1	-1	-1	1	-1	-1	1	-1
P2	1	-1	1	-1	1	1	1	-1
P3	1	-1	-1	1	1	1	-1	-1
P4	1	-1	1	1	1	1	1	-1
P5	-1	1	1	-1	-1	-1	-1	-1
P6	1	-1	-1	1	-1	1	-1	-1
P7	1	-1	-1	1	1	1	1	-1
P8	1	-1	1	-1	-1	1	1	-1
P9	1	1	1	1	1	1	1	-1

a)

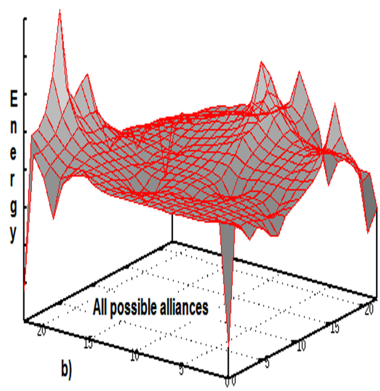

Fig. 1 a) Nine political parties and positions about eight social aspects. b)Landscape for all possible alliances

In mathematical terms, the alliance ensemble problem is stated as follows. Let A be the group of N agents and A_i the description of agent i in terms of its features (vector) and C the optimal alliance ensemble, which consists of at least two disjoint alliances. The ensemble contains alliances C_j, $j > 1$, that is, by definition there is more than one alliance and each agent A_i belongs to one and only one of them.

The optimal alliance ensemble is defined as:

$$C = minarg \sum_{i,k} \delta(A_i, Ak) \tag{4}$$

where $\delta(A_i, A_k)$ are the differences between agent A_i and agent A_j over all the feature vectors, measured as the distance between vectors A_i and A_j.

Each agent is defined through a set of variables and each agent may change one or more of these variables over time. Agents try to interact with each other in order to devise better states, that is, better alliances. From the statistical mechanics point of view, the landscape defined by all the possible ensembles may have several local minima, but also may have a global minimum and thus, the ensemble be optimal.

Axelrod proposed this model, based on spin glass theory [4], as an alternative explanation to alliance formation in European countries in the Second World War. There was a global optimum consisting of two alliances which were very similar to the ones observed during the conflict. Although the landscape model is very interesting from the sociophysics point of view [20, 57], it is not feasible in most human alliances as it requires absolute information access for all agents, being humans, parties, companies or any other social structure. In the great majority of cases, agents are not allowed to be aware of the state and features that describe other agents.

4.2 Algorithms

Several alliance formation algorithms have been proposed in the context of multiagent systems [49, 29, 50]. The main idea is to maximize the sum of the payoffs to all the alliances by identifying the optimal combination of alliances and the division of agents into these alliances. Each agent has a set of tasks that may or may not be similar to the tasks assigned to other agents. The general alliance formation algorithm can be summarized as:

1. Construct a list of possible alliances, $S_i(q)$ of up to q agents.
2. While $S_i(q)$ is not empty, do:
3. Contact agents $A_j \in S_i(q)$.
4. Evaluate the benefit of joining A_j in an alliance, subject to preferences and constraints.
5. Extract q from $S_i(q)$ and share subtasks.
6. If contacted by agent A_k, substract q as well as the common tasks.

Alliance formation algorithms may be classified according to the general assumptions of communication and information access for individual agents. The first class is that of complete information assumptions, which includes dynamic programming algorithms that guarantee to find the optimal alliance, but whose complexity is prohibitive for real applications [62, 42, 30]. In this scheme, the messages that are sent between agents in order to share their preferences could be exponential, although some alternatives have been proposed to reduce complexity. The second class is that of heuristic-based algorithms that do not guarantee finding the optimal, but reach solutions very fast [45, 61].

Coalition formation in human societies does not follow the dynamics specified by many of the known alliance formation algorithms, though alliances still tend to be robust (at least for a short period of time). In human societies, agents are unreliable, do not completely share information, and may even lie in order to obtain better profits [22]. Alliance formation is thus based on a weak communication scheme, i.e., agents do not completely share their information.

As a consequence of the lack of pervasive information, each agent is forced to explore its local environment. Agents, therefore, have to join an alliance based solely on local information and a limited knowledge of the features of other agents in the alliance.

Human alliances are guided by heuristics that do not always lead to robust ensembles, i.e., the alliances may not correspond to a global optimum. However, some human alliances tend to be robust and long lasting [21]. It is of special interest to study the general behavior of agents in this context as a source of inspiration for new algorithms that do not need all the available information.

From Political and Management Sciences as well as from Sociology, there are several successful examples of human alliance formation, at least for the allies' immediate purposes. However, the grounds for those alliances are not always obvious from the analysis of the agents, and, in some cases, the alliance might even appear contradictory in itself. In general, social agents are not homogeneous. They are often expressed with weighted influence of some over others or with non-linear relations among them. For example, agent i may be in the same alliance as agent j, as long as agent k is also part of it. However, agent k may be inclined to form an alliance with agent j as long as agent i is not part of it [21].

In human alliance formation, dynamics are mainly governed by two types of coordination. The first one is a centralized organization where agents communicate with agents from other groups via their group as a whole [42]. The second coordination scheme is decentralized, i.e., there is no group of agents that is pervasive in the environment and therefore, the coordination must be achieved through individual interactions.

Based on the above-mentioned constraints and features of human alliances, several coalition algorithms have been proposed. In [37], the agents do not have knowledge of the complete feature vector of all agents, leading to restricted alliance formation. A set of agents is a restricted alliance when all agents agree to share information within the set, but not with the rest of agents. [38, 41] describe a multiagent system model of humanitarian assistance services in conflict areas, in which agents represent non-government organizations that, in general, have a common objective. The available information is not always reliable and the agents are aware of this fact when they decide whether or not to group with another agent.

The variety of reasons why an individual may have to become part of an alliance have been included in some schemes in order to obtain stable alliances when complete information is not available. For example, in [12] each agent i is provided with a variable that quantifies its strength of character determining its predisposition to form complete new alliances, C_i. A second variable G_i defines its attraction to obtain a profit within that alliance. Finally, a third variable R_i is defined that states its reluctance to abandon its current alliance to join another one.

Agents with a high C_i will send invitations to form new coalitions more often that those with lower predisposition to form new aliances. When an agent i that is part of a coalition A_{act} is invited to join another alliance A_n, its decision is based not only on the benefit x_i^n it will receive but also on the individual parameters that define its *personality*. The gains from both the new and actual coalition, $S(A_n)$ and $S(A_{act})$, are determined by the agent through the benefits it would achieve and from individual preferences:

$$S(A_{act}) = G_i \times \frac{x_i^{act}}{x_i^{act} + x_i^n} + R_i \times (1 - sb_{act}) \tag{5}$$

$$S(A_n) = G_i \times \frac{x_i^n}{x_i^{act} + x_i^n} + R_i \times (1 - sb_n) \tag{6}$$

where sb_{act} is a parameter that summarizes the stability of the coalition that contains agent i, and sb_n is the stability of the new coalition. So, if $S(A_{act}) \leq S(A_n)$, the agent decides to abandon alliance A_{act} and be part of the alliance A_n. When agents are homogeneous ($G_i = R_i = C_i$, $\forall i$), the simulations converge to a unique stable structure, but when the personality is heterogeneous, several coalitions may result. As each agent's decision is influenced not only by the profit it will obtain by joining an alliance but also by its own personality, non-optimal alliances are avoided.

5 Optimization through Social Labeling

Classification of individuals in a society is a common phenomenon. The label or tag assigned to an individual may be the result of prejudices, ignorance or true facts. An individual's tag may influence the way other individuals interact with him/her, by inducing a positive or negative reaction/feeling [26]. Some algorithms have been inspired by such social tags.

Achieving cooperation in Peer to Peer networks (P2P) is not a trivial task. Some peers may decide to stop cooperating once they have obtained the resources they were looking for, as often happens in file-sharing schemes. A P2P network in which the number of attempts made to obtain a given resource for all peers is minimized is highly desirable. In [25], a method is presented that improves cooperation in P2P networks based on the evaluation among peers leading to each individual being assigned a tag. Peers that tend to avoid cooperation are tagged as non-cooperative whereas peers that tend to cooperate are tagged positively. The algorithm is:

1. While the number of generations is not met:
2. for each agent i in the population:
3. Select a game partner agent j with a similar tag (whenever possible).
4. Peers i and j interact through their strategies and get payoff.
5. Reproduce agents proportionally to their payoff.
6. Mutate tags and strategies of each reproduced agent.

Each peer is assigned a strategy that states its behavior for interacting with other agents. This strategy is based on the Prisoner's Dilemma (PD), as proposed in [4]. Two agents (peers) are involved in a situation where they have the option to cooperate (C) with each other or to defect (D). Each agent obtains a payoff as a function of its action and the action of the other peer. The payoff proposed in the PD is as follows:

$$T > R > P > S \tag{7}$$

$$2R > T + S \tag{8}$$

where T is the payoff an agent receives if it defects and the other cooperates, R is the payoff when both agents cooperate, P is the payoff when both of them defect, and S is the payoff an agent receives if it cooperates and the other defects. The second condition is to prevent an agent from alternating between cooperation and defection.

As it is known in game theory, if there is only one encounter between two agents, the best strategy is to defect (D). In the proposed model there is no possibility of confronting the same peer twice, which makes the result surprising: the best strategy, in terms of a cooperative network, is that in which cooperative peers (C) are dominant. In other words, the overall number of attempts for each peer to obtain a given resource is minimized through cooperation.

Optimization is achieved by choosing a more convenient counterpart (step 3 in the previous algorithm) in order to have a significant level of cooperation [25]. When tags are removed, the achieved network is highly inefficient, as the peers do not obtain the desired resource because of the lack of cooperative agents. An alternative is to find an outstanding member within the society that performs better than the rest as in [43].

Another algorithm based on social tags is that presented in [2]. Here, a mining newsgroup algorithm leads to the classification of people in two opposite camps over a discussion issue. The algorithm is based on the assumption that people respond more frequently to messages that contain ideas they do not agree with. Through their responses, people get tagged and this tag is taken into account to define the topology of a bipartite network in which each vertex represents a participiant and edges (E) represent responses between participants.

The algorithm seeks a partition of the vertices into two sets: F and A, one representing participants in favor and the other representing users against the discussion issue. The central hypothesis is: if most edges in the graph associated to the newsgroup represent disagreement then the optimum choice of F and A maximizes $f(F,A)$, which is the cut function: $f(F,A) = |E \cap (F \times A)|$). It is only possible to seek this bipartition when the opinions are tagged (favor, against). The mining algorithm, through tags, obtains better results than those obtained by statistical analysis of texts, which is the most used algorithm.

Tags may lead to a separation of cooperative and non-cooperative peers as a side effect in P2P networks and discrimination against participants in debates. However, the richness of segregation has been explicitly studied in several algorithms that are presented in the next section.

6 Neighborhood Delimitation and Segregation

Although house formation and house-keeping processes exist in termites, ants, birds and other species [14], the complexity and richness of human neighborhood formation and segregation is immense. The genesis, evolution and structure of cities are the result of several factors, some external and some endogenous. Economic rules, political constraints and psychological factors, among many other factors determine the overall distribution of neighbors over a city [32, 7].

Residential segregation and neighborhood delimitation has been widely studied from many perspectives within the social sciences [33, 28, 39]. There are at least two approaches to this process. The first is that of phenomenological analysis, as in the previous cited works, and the second is a constructive approach based on multiagent models. One of the early examples of the later approach is that of [46]. In this model, the consequences of many individual decisions and its counterintuitive foundations in urbanism were presented and the dynamics underlying the city genesis and evolution were subject of formal analysis.

The segregation behavior states that all householders try to live near people very similar to them and avoid householders radically different from them. Each householder is described by a feature vector. In mathematical terms, each householder H_i has a propensity to stay comfortably with his/her neighbors defined by the total differences between H_i and H_k, where k are the neighbors of H_i.

In the Schelling model (SM), each agent occupies a cell in a rectangular lattice. The segregation of dissimilar agents is guided by a simple set of rules. At the beginning, agents are randomly distributed in the lattice and after a few decisions by each agent, a structured (segregated) distribution appears [53]. This process is an example of self-organization, a common feature present in several processes, among them social phenomena. Each agent A_i is subject to the following rules:

1. Compute the fraction of neighbors that are of the same color, that is, that have a similar feature vector, F_i.

2. If the agent is satisfied with its neighbors, then it stays in its actual location: $F : i > T_i$, where T_i is the satisfaction threshold. If this condition is satisfied, then the agent ends.

3. The agent looks for the nearest available location that satisfies its requirements and moves there.

The Schelling model has been widely studied as a theoretical tool to explain segregation phenomena not only in the cities, but also in international conflicts [60]. Besides these applications in social sciences, SM has also been studied in other contexts. It has inspired several algorithms in network routing. For example, in [52] a Schelling-resembling algorithm dynamically modifies the topology of a network of hubs, and in [51], it manages to improve bandwidth in P2P networks.

P2P networks are distributed and thus, present a lack of any central control. All nodes present the same functionality and peers can cooperate and communicate with each other based only on a virtual communication network with the constraint that peers are aware only of their local topology.

In a P2P network the topology is dynamic, i.e., peers in contact may decide to delete the link or peers who were previously not in contact may create a new link. In this kind of network, nodes have a maximum number of connections and when the network grows, some algorithms tend to form hubs, i.e., nodes with a higher number of connections than the rest of the nodes. This may lead to several communication problems. The Schelling abstract algorithm (SAA) is a topology modification scheme that not only prevents the emergence of hubs, but also leads to topologies with high cohesion [54]. Each peer has a desired percentage of neighbors with similar properties, $PNSP_{des}$, that it tries to preserve. The general algorithms is:

1. set the $PNSP_{des}$.

2. while true

3.   obtain $PNSP_{act}$

4.   if $PNSP_{act} < PNSP_{des}$

5. drop a neighbor with a different property.
6. search for another peer with more similarity.

Peers are described in terms of their number of connections, their bandwidth and other features. When the actual percentage of peers with similar properties with whom peer i is linked to (its neighborhood), $PNSP_{act}$, is lower than $PNSP_{des}$, peer i deletes one link. The link to one of the most different peers is the one chosen for deletion. Then, peer i starts its search for another peer with some resemblance to it.

The SAA segregates peers based on their descriptions leading to well structured networks in which the achieved topologies are connected, which is a very desirable feature in P2P networks [63]. As $PNSP_{des}$ increases, the algorithm leads to networks with a higher number of segregating clusters and at a critical point, the network may become unconnected.

7 Further Horizons and Conclusions

In this section, as a part of a wider panorama into algorithms inspired by social phenomena, we propose some ideas that may contribute to enrich the existing options in the Computer Science community.

Leadership is presented as an influence model in which the best individuals are followed by others. However, there are plenty of social phenomena [14] that are decentralized and distributed, i.e., there is not a leader at all. Human behavior such as that of opinion dynamics has attracted attention from the computer science community [5, 20] as it is stated that opinions may not be influenced (at least directly) by leaders opinion.

The assumption that leaders are the best solutions is not entirely true in more realistic approaches because there is a wide range of individuals with some responsibilities that may be interesting to study. Among them is the negotiator, which is responsible for communicating with other leaders and transmitting to them the desires of his/her own leader. A negotiator is a character that could make the work of leaders a lot easier.

In leader-based algorithms, in which migration is important, human patterns of mobility at local [23] or global scales [59] have not been widely considered. A common strategy of exploring unknown spaces in mammals, including humans, is that of Lévy flight [13], which has been incorporated into some optimizing algorithms, such as those of [24], which tend to copy the outstanding foraging strategy of many species.

The self-organizing process of neighborhood segregation is very simple and very elegant. However, there are other aspects of neighborhood delimitation and segregation that are being considered in more detailed models, for example, a wider neighborhood of influence or a more detailed preference for each agent [7, 8]. Algorithms inspired by these behaviors may be applied in, for example, cluster formation.

Most of the algorithms whose origin is a metaphor of social phenomena obtain results equivalent to algorithms inspired by other metaphors. Although the results are encouraging, a lot of ideas and metaphors from human social phenomena are still waiting for further exploration.

References

1. Larisa, A.: Como sobreviven los marginados. Siglo XXI editores, México (1975)
2. Agrawal, R., Rajagopalan, S., Srikant, R., Xu, Y.: Mining newsgroups using networks arising from social behavior. In: Proceedings of the 12th International Conference on WWW, pp. 529–535 (2003), doi:10.1145/775152.775227
3. Aknine, S.: A reliable algorithm for multi-agent coalition formation. In: Proceedings of the IEEE International Symposium on Intelligent Control/Intelligent Systems and Semiotics, Cambridge, MA, USA, pp. 290–295 (1999), doi:10.1109/ISIC.1999.796670
4. Axelrod, R.: The evolution of cooperation. Basic Books (1984)
5. Bagnoli, F., Franci, F., Rechtman, R.: Opinion formation and phase transitions in a probabilistic cellular automaton with two absorbing states (2005) (accessed July 25, 2008), `http://arxiv.org/abs/nlin/0511001`
6. Ball, P.: Critical mass. Farrar, Straus and Giroux, London (2006)
7. Batty, M.: Cities and complexity. MIT Press, Cambridge (2007)
8. Benenson, I.: Multiagent simulations of residential dynamics in the city. Computers, Environment and Urban Systems 22(1), 24–42 (1998)
9. Bonabeau, E., Dorigo, M., Theraulaz, G.: Inspiration for optimization from social insect behaviour. Nature 406, 39–42 (2000)
10. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: From natural to artificial systems. Oxford University Press, Oxford (1999)
11. Bonabeau, E.: Agent-based modeling: Methods and techniques for simulating human systems. PNAS 99, 7280–7287 (2002)
12. Bonnevay, S., Kabachi, N., Lamure, M.: Agent-based simulation of coalition formation in cooperative games. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Rechnology, pp. 136–139 (2005), doi:10.1109/IAT.2005.33
13. Boyer, D., Miramontes, O., Ramos-Ferández, G., Mateos, J., Cocho, G.: Modeling the searching behavior of social monkeys. Physica A 342, 329–335 (2004)
14. Camazine, S., Deneubourg, J., Franks, N., Sneyd, J., Theraulaz, G., Bonabueau, E.: Self-organization in biological systems. Princeton University Press, Princeton (2003)
15. de Castro, L.: Fundamentals of natural computing. Chapman & Hall, Boca Raton (2007)
16. Coello, C.A.C., Landa, R.: Evolutionary multiobjective optimization using a cultural algorithm. In: Proceedings of the IEEE Swarm Intelligence Symposium, pp. 6–13 (2003), doi:10.1109/SIS.2003.1202240
17. Daneshyari, M., Yen, G.: Talent based social algorithm for optimization. In: Proceedings of the Congress on Evolutionary Computation, pp. 786–791 (2004), doi:10.1109/CEC.2004.1330939
18. Das, S., Flocchini, P., Kutten, S., Nayak, A., Santoro, N.: Map construction of unknown graphs by multiple agents. Theoretical Computer Science 385, 34–48 (2007)

19. Flake, G.: The computational beauty of nature. MIT Press, Cambridge (1998)
20. Florian, R., Galam, S.: Optimizing conflicts in the formation of strategic alliances (2004) (accessed Febraury 20, 2008), http://arxiv.org/abs/cond-mat/0004216v1
21. Gil, J., Schmidt, J.: La red de poder en México. UNAM-IIMAS, México (1999)
22. Gil, J., Schmidt, J., Castro, J., Ruiz, A.: A dynamic analysis of the Mexican network of power. Connections 20, 34–55 (1997)
23. González, M., Hidalgo, C., Barabasi, A.: Understanding individual human mobility patterns. Nature 453, 779–782 (2008)
24. Gutowski, M.: Lévy flights as an underlying mechanism for global optimization algorithms (2001) (accessed June 10, 2008), http://arxiv.org/abs/math-ph/0106003v1
25. Hales, D., Edmonds, B.: Applying a socially inspired technique (tags) to improve cooperation in P2P networks. IEEE Transactions on Systems, Man and Cybernetics Part A 35(3), 385–395 (2005)
26. Holland, J.: The effect of label (tags) on social interactions. SFI Working Paper 93-10-064 (1993) (accessed January 19, 2008), http://www.santafe.edu/sfi/publications/wplist/1993
27. Horling, B., Lesser, V.: A survey of multi-agent organizational paradigms. The Knowledge Engineering Review 19(4), 281–316 (2005)
28. Jargowsky, P.: Take the money and run: Economic segregation in US metropolitan areas. American Sociological Review 61(6), 984–998 (1996)
29. Kijima, K.: Agent-based simulation of alliance formation and its stability analysis: Application to aviation industry (2003) (accessed May 18, 2008), http://aisel.aisnet.org/pacis2003/81
30. Klush, G.: Dynamic coalition formation among rational agents. IEEE Intelligent Systems 17, 42–47 (2005)
31. Kohonen, T.: Self-organizing maps. Springer, Heidelberg (2000)
32. Kotkin, J.: The city, a global history. Weidenfeld & Niolson (2005)
33. van der Laan, W.: Involuntary isolation: Ethnic preferences and residential segregation. Journal of Urban Affairs 29(3), 289–309 (2007)
34. Langaney, A., Clottes, J., Guilaine, J., Simmonet, D.: La plus belle histoire de l'homme. Editions du Seuil, Paris (1998)
35. Lewontin, R.: It ain't necessarily so: The dream of the human genome and other illusions. New York Review of Books (2000)
36. MacKay, D.: Information theory, inference and learning algorithms. Cambridge University, Cambridge (2003)
37. Mashkov, V.: Restricted alliance and coalitions formation. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, pp. 329–332 (2004)
38. Mashkov, V., Marik, V.: Alliance formation process and communication traffic (2002) (accessed February 10, 2008), http://www.actapress.com/PaperInfo.aspx?PaperID=25670&reason=500
39. Massey, D., Gross, A., Shibuya, K.: Migration, segregation and the geographic concentration of poverty. American Sociological Review 59(3), 425–445 (1994)
40. Nielsen, J.: The myth of leadership: Creating leaderless organizations. Davies-Black Publishing (2004)

41. Pechoucek, M., Marik, V., Barta, J.: A knowledge-based approach to coalition formation. IEEE Intelligent Systems 17(3), 17–25 (2002)
42. Rahwan, T., Jennings, N.: An improved dynamic programming algorithm for coalition structure generation. In: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multi-Agent Systems, International Foundation for Autonomous Agents and Multiagent Systems, pp. 1417–1420 (2008)
43. Ray, T., Liew, M.: Society and civilization: An optimization algorithm based on the simulation of social behavior. IEEE Transactions on Evolutionary Computation 7(4), 386–396 (2004)
44. Reynolds, R.: An introduction to cultural algorithms. In: Sebald, A.V., Fogel, L.J. (eds.) Proceedings of the 3rd Annual Conference on Evolutionary Programming, pp. 131–139. World Scientific Press, Singapore (1994)
45. Sen, S., Dutta, P.: Searching for optimal coalition structures. In: Proceedings of the 4th International Conference on Multiagent Systems, pp. 286–292 (2000)
46. Schelling, T.: Micromotives and Macrobehavior. Ed Norton (2006)
47. Selvakumar, A., Thanushkodi, K.: Optimizing using civilized swarm: Solution to economic dispatch with multiple minima. Electric Power Systems Research 79(1), 8–16 (2009)
48. Semsar-Kazerooni, E., Khorasani, K.: Optimal consensus algorithms for cooperative team of agents subject to partial information. Automatica 44(11), 2766–2777 (2008)
49. Shehory, O., Kraus, S.: Task allocation via coalition formation among autonomous agents. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, pp. 655–661 (1995)
50. Shehory, O., Kraus, S.: Formation of overlapping coalitions for precedence-ordered task-execution among autonomous agents. In: Proceedings of the 2nd International Conference on Multi-Agent Systems, pp. 330–337. AAAI Press/MIT Press (1996)
51. Singh, A.: Topology adaptation in p2p networks using Schelling model. The University of Dublin Trinity College (2004) (accessed March 21, 2008), `https://www.cs.tcd.ie/publications/tech-reports/reports.04/TCD-CS-2004-41.pdf`
52. Singh, A., Haar, M.: Creating an adaptive network of hubs using Schelling model. Communications of the ACM 49(3), 69–73 (2006)
53. Singh, A., Vainchtein, D., Weiss, H.: Schelling's segregation model: Parameters, scaling, and aggregation (2007) (accessed July 20, 2008), `http://arxiv.org/abs/0711.2212v1`
54. Singh, A., Haahr, M.: Decentralized clustering in pure P2P overlay networks using Schelling's Model. In: Proceedings of the IEEE International Conference on Communications, pp. 1860–1866 (2007), doi:10.1109/ICC.2007.310
55. Ursem, R.: Multinational evolutionary algorithms. In: Proceedings of the Congress on Evolutionary Computation, Washington, DC, USA, pp. 1633–1640 (1999)
56. Vijaya, P., Murty, N., Subramanian, D.: Leaders-subleaders: An efficient hierarchical clustering algorithm for large data sets. Pattern Recognition Letters 25, 505–513 (2004)
57. Wan, Y., Wu, H.: Convergence and optimization of agent-based coalition formation. Physica A: Statistical Mechanics and its Applications 348, 641–658 (2004)
58. Wang, J., Wang, D.: Particle swarm optimization with a leader and followers. Progress in Natural Science 18(11), 1437–1443 (2008)
59. Weaver, T., Roseman, C.: New developments in the genetic evidence for modern human origins. Evolutionary Anthropology 17, 69–80 (2008)

60. Yamamoto, K.: International relations and agent-based modeling. Paper presented at the Annual Meeting of the International Studies Association 48th Annual Convention, Chicago, IL, USA (10-09-2008) (2007),
http://www.allacademic.com/meta/p180204_index.html
61. Yang, S., Su, C.: Co-evolutionary learning with strategic coalition for multiagents. Applied Soft Computing 5, 193–203 (2005)
62. Yeh, D.: A dynamic programming approach to the complete set partitioning problem. Numerical Mathematics 26(4), 467–474 (1986)
63. Zhang, Y., Li, Y., Yue, Y., Zhu, L.: An ecology-based model for efficient distributed search in P2P networks. In: Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China, pp. 4485–4488 (2008), doi:10.1109/WCICA.2008.4593645
64. Zomaya, A.: Handbook of nature-inspired and innovative algorithms. Springer, Heidelberg (2006)