Nature-inspired Decentralized Cooperative Metaheuristic Strategies for Logistic Problems

Belén Melián Batista, José A. Moreno Pérez and J. Marcos Moreno Vega
GIC, Group of Intelligent Computing
IUDR, Instituto Universitario de Desarrollo Regional
Universidad de La Laguna
E-38271 La Laguna, Spain
Phone: +34-922-318186, Fax: +34-922-318170
email: {mbmelian,jamoreno,jmmoreno}@ull.es
Research Projects TIN2005-08404-C04-03 and PI042005/044

ABSTRACT: Cooperative metaheuristics are those based on several search processes that interchange information while the search is being developed. In a decentralized cooperative strategy, each search process has its own rules to decide when and how interchange the relevant information with other processes. Several of the most important cooperative strategies for metaheuristic solution procedures are nature-inspired. The information exchanged by the processes is very important in the success of the search process. The selection of this information depends not only on the cooperation strategy but also on the instances of the problems at hand; it must be relevant and effective. We analyse the information exchanged in the application of these strategies to logistic optimization problems. They include vehicle routing problems, loading problems and location problems with the corresponding standard problems: the Travelling Salesman Problem, the Bin Packing Problem and the p-Median Problem, respectively.

KEYWORDS: Decentralized Cooperative Strategies, Metaheuristics, Logistic Problems.

INTRODUCTION

The metaheuristicis are general strategies to design heuristic algorithms to solve some optimization problems. The cooperative metaheuristics are those based on several search procedures that interchange relevant information on the solutions found and the search processes while the search is being developed. Each one of these search procedures ranges form the implementation of a very simple local search up to any sophisticated metaheuristic technique. A cooperative strategy consists of the set of rules that state the way to conduct the cooperation between search procedures in a cooperative solution method. In the centralized cooperative strategies there is a central agent that conducts the interchange of information between the processes. In the other side, the decentralized cooperative strategies are those where each search process has its own rules to decide when and how interchange the relevant information with other processes. Several of the most important centralized and decentralized cooperative strategies for metaheuristic solution procedures are nature-inspired. The main examples of decentralized cooperative strategies are nature-inspired like those based on Genetic Algorithms, Ant Colony Systems and Particle Swarm Optimization.

The influence of nature in computing started already in its early days when Von Neuman used the brains as inspiration for his model of a computer. Among several lines of research, the biology has been used as a metaphor for the designing of algorithms [29], [16].

However, the information exchanged by the processes is very important in the success of the search procedures. The selection of this information depends not only on the cooperation strategy but also on the instances of the problems at hand; it must be relevant and effective. We analyse the information exchanged in the application of these strategies to solve large real instances of the main logistic problems. The logistic optimization problems include vehicle routing problems, loading problems and location problems. The corresponding standard problems in these classes of problems are among the most studied NP-hard problems in combinatorial optimization: the Travelling Salesman Problem, the Bin Packing Problem and the p-Median Problem, respectively. On the other hand, the problems that appear in real circumstances are often mixed problems and usually include some new constraints or characteristics that motivate the need of robust optimization procedures adaptable to the new situations.

The metaheuristics are the most adaptable solution procedures for hard optimization problems and the cooperative strategies provide the desired robustness by including several kinds of search techniques in the cooperating procedures and can be efficiently parallelized to improve their performance with decentralized parallel architectures. The decentralized cooperation processes found in the natural world are good sources of inspiration for the design of cooperative strategies. The corresponding metaheuristics are promising methods for approaching the large real instances of the most important logistic problems.

Next sections review some of the cooperation behaviour found in nature and how they inspired artificial systems. We concentrate in decentralized cooperation that is used to design decentralized cooperative metaheuristic searches for optimization problems. The last sections briefly reflect the relevance of these methods in Logistic optimization and provide some conclusions.

COLLECTIVE INTELLIGENCE FROM NATURE

Cooperation is a ubiquitous phenomenon in nature. The studies show an, each time less surprising, intelligent collective decision in sets of very simple elements. The corresponding collaborative ways for performing a common task have been used as inspiration for useful artificial systems.

COLLECTIVE INTELLIGENCE

Collective intelligence [1] is the intelligence of a collective, which emerges from the interaction of many individuals and the intelligences of these individuals. Collective intelligence appears in a wide variety of forms of cooperative decision making in quarks, organisms, animals, humans, and computers. The study of collective intelligence may properly be considered a subfield of Sociology, Behavioural Ecology, Ethologic, Anthropology, Economy, Computer science, and of mass behaviour--a field that studies collective behaviour from the level of quarks to the level of bacterial, plant, animal, and human societies. Decentralized Cooperative strategies found in nature have been used for designing computational tools for solving problems.

Cooperation is a ubiquitous phenomenon in nature. We say that there is cooperation between a series of items when there is an interaction among them that allows the achievement of some goals. Within living organisms there is cooperation at many different levels. There is cooperation between chromosomes, proteins, cells, tissues and organs that allows the nowadays life. There is cooperation in the behaviour of a swarm of insects, a school of fish, a group of wolves, a herd of lions that is useful for survivability of the specie. The existence of cooperation is observed not only within some species but also sometimes between individuals of different species that do different tasks; this collaboration allows their adaptability to the environment. There is a cooperative system when the agents of the system work together using some cooperation mechanism to achieve a goal. Human society is probably the largest cooperative system in nature. We use very different mechanisms of cooperation at different levels, at different scopes and at the same time.

Theoretical and experimental studies have suggested that self-organization constitutes a plausible explanation to the success of some collective tasks without reference to any kind of central control or regulation. The correspondence between individuals and search processes provides a way to design nature-inspired cooperative strategies for metaheuristics. When the cooperation among the interacting agents is not subject to a kinship relationship or any kind of central control or regulation the resulting strategy will inspire a decentralized cooperation metaheuristic. In the field of decentralized cooperation strategies from nature, the cooperating entities are named “chromosomes”, “cells”, “neurons”, “bacteria”, “membranes”, “ants”, “particles”, “bees”, or “individuals”. For inspiring cooperative strategies in metaheuristic development, we usually find social interactions in cooperative problem solving where several autonomous individuals work together to achieve some goal.

SWARM INTELLIGENCE

The expression "swarm intelligence" [8] was introduced by Beni & Wang in 1989, in the context of cellular robotic systems. The ecological success of ants, and some other social insects, has provided inspiration for the design of several metaheuristic optimization procedures. Swarm Intelligence systems are typically made up of a population of simple agents interacting locally with one another and with their environment. Although there is normally no centralized control structure dictating how individual agents should behave, local interactions between such agents often lead to the emergence of global behaviour. Examples of systems like this can be found in nature, including ant colonies, bird flocking, animal herding, bacteria moulding and fish schooling. Some phenomena studied for that purpose have been the organization of collective foraging activities, the division of labour in the colony, the construction of tunnelling networks, the clustering of brood items and the collective building processes. [26]

Theoretical and experimental studies have suggested that self-organization constitutes a plausible explanation to the success of some collective tasks without reference to any kind of central control or regulation. Social colonies are collectively intelligent individuals that may be very simple and lack the sort of intelligence that only emerges at the global level. Self organization is the spontaneous emergence of a global order from local interactions and some degree of randomness at the individual level [50]. Several models of collective behaviours in social insects have provided inspiration for effective metaheuristic procedures to solve problems. These artificial swarm intelligent systems have successfully applied to many optimization problems in logistics

ANT COLONY OPTIMIZATION

The Ant Colony Optimization (ACO) [24] studies artificial systems that take inspiration from the behavior of real ant colonies in finding shortest paths to food. The ants communicate by the concentration of pheromone in the edges used in the path. So the information used by the ants is related with the number of times a given edge is in a good solution. In [23] the Ant Colony Optimization metaheuristic was formally defined. The first paper by Dorigo et al. on ACO [21] was later published with the title "Optimization by a colony of cooperating agents" [22] which reveals the relevance of this metaheuristic in the field of nature-inspired cooperative strategies.

The number of applications of ACO algorithms has increased very strongly over the recent years and ACO has been applied in the meantime to certainly more than one hundred different problems [19]. Among these problems we found several relevant problems in Logistics. The ACO method mimics the behaviour of the ants with artificial ants that are continuously walking around a graph representing the problem. This is important in changing environment where the instances data varies slightly and the ants can be running continuously and adapting to the changes in real time. The ACO algorithms have been successfully used to solve several routing problems like VRP (Vehicle Routing Problems) and TSP (Travelling Salesman Problem).

STOCHASTIC DIFFUSION SEARCH

Stochastic Diffusion Search (SDS) [17] is another generic optimization search method inspired in the tandem running procedure observed in some species of ant. The SDS is best suited to problems where the objective function can be decomposed into multiple independent partial-functions. Each SDS agent maintains a hypothesis which is iteratively tested by evaluating a randomly selected partial objective function parameterised by the agent's current hypothesis. In the standard version of SDS such partial function evaluations are binary resulting in each agent becoming active or inactive. Information on hypotheses is diffused across the population via inter-agent communication. Unlike the stigmergetic communication used in ACO, in SDS agents communicate hypotheses via a one-to-one communication strategy analogous to the tandem running procedure observed in some species of ants [30]. A positive feedback mechanism ensures that, over time, a population of agents stabilise around the global-best solution. SDS is both an efficient and robust search and optimisation algorithm, which has been extensively mathematically described.

PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization (PSO) [11] is another form of swarm intelligence. It is a metaheuristic optimization method inspired by social behaviour of bird flocking or fish schooling that appeared in 1995 [38]. It is a population based procedure where the potential solutions, called particles, fly through the problem space by following the current optimum particles. Imagine a swarm of insects or a school of fish. If one sees a desirable path to go (e.g., for food, protection, etc.) the rest of the swarm will be able to follow quickly even if they are on the opposite side of the swarm. On the other hand, in order to facilitate exploration of the search space, each particle must have a certain level of randomness in its movements, so that the movement of the swarm has a certain explorative capability: the particle should be influenced by the rest of the swarm but also should independently explore to a certain extent. This is a manifestation of the basic exploration-exploitation tradeoffs that occur in any search problem.

In past several years, PSO has been successfully applied in many research and application areas. It is demonstrated that PSO gets better results in a faster, cheaper way compared with other methods.

The nature is modelled by particles in multidimensional space that have a position and a velocity. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. Another value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbours of the particle. Therefore, these particles are flying through hyperspace and have two essential reasoning capabilities: their memory of their own neighbourhood best position and knowledge of the swarm's best, "best" simply meaning the position with the smallest objective value. Members of a swarm communicate good positions to each other and adjust their own position and velocity based on these good positions. There are two main ways this communication is done: a global best position that is known to all and immediately updated when a new best position is found by any particle in the swarm and "neighbourhood" best positions where each particle only immediately communicates with a subset of the swarm about best positions. Neighbourhood bests allow better exploration of the search space and reduce the susceptibility of PSO to falling into local minima, but slow down convergence speed. Note that neighbourhoods merely slow down the proliferation of new bests, rather than creating isolated sub-swarms because of the overlapping of neighbourhoods. Smaller neighbourhoods lead to slower convergence, while larger neighbourhoods to faster convergence, with a global best representing a neighbourhood consisting of the entire swarm.

Repulsive particle swarm optimization (RPSO) [51] is a variant of particle swarm optimization characterized by the existence of repulsion between particles. This feature allows preventing the swarm being trapped in local minima, which would cause a premature convergence and would lead the optimization algorithm to fail to find the global optimum. The main difference between PSO and RPSO is the propagation mechanism to determine new positions for a particle in the search space. RPSO is capable of finding global optima in more complex search spaces. On the other hand, compared to PSO it may be slower on certain types of optimization problems.

BACTERIAL SWARM INTELLIGENCE

Bacterial Swarm Intelligence (BSI) [28] is based on the behaviour of some colonies of bacteria. Under natural growth conditions, bacteria can utilize intricate communication capabilities (e.g. quorum-sensing, chemotactic signalling and plasmid exchange) to cooperatively apply a self-organization strategy to elevate their adaptability; the colonial pattern is collectively engineered according to the encountered environmental conditions. Instead of genetically store all the information required for the design of the strategies, the bacterial colony cooperatively generates the additional information needed as required for the self organization strategy. It seems that the bacteria use their intracellular flexibility, involving signal transduction networks and genomic plasticity, to collectively maintain linguistic communication: self and shared interpretations of chemical cues, exchange of chemical messages and dialogues. Meaning-based communication permits colonial identity, intentional behaviour (e.g. pheromone-based courtship for mating), purposeful alteration of colony structure (e.g. formation of fruiting bodies), decision-making (e.g. to sporulate) and the recognition and identification of other colonies. These features are associated with a bacterial social intelligence. Such a social intelligence would require going beyond communication to encompass unknown additional intracellular processes to generate inheritable colonial memory and commonly shared genomic context.

Complex colonial forms (patterns) emerge through the communication-based singular interplay between individual bacteria and the colony. Each bacterium is, by itself, a biotic autonomous system with its own internal cellular informatics capabilities (storage, processing and assessment of information). These afford the cell plasticity to select its response to biochemical messages it receives, including self-alteration and the broadcasting of messages to initiate alterations in other bacteria. Hence, new features can collectively emerge during self-organization from the intracellular level to the whole colony. The cells thus assume newly co-generated traits and abilities that are not explicitly stored in the genetic information of the individuals.

Cooperation among individuals is necessary for evolutionary transitions to higher levels of biological organization. In such transitions, groups of individuals at one level (such as single cells) cooperate to form selective units at a higher level (such as multicellular organisms). Though the evolution of cooperation is difficult to observe directly in higher eukaryotes, microorganisms do offer such an opportunity. The evolution of novel cooperative behaviour in experimental lineages of the bacterium Myxococcus xanthus have been reported. Wild-type strains of this bacterium exhibit socially dependent swarming across soft surfaces by a mechanism known as ‘S-motility’ that requires the presence of extracellular pili. Evolved swarming is mediated, at least in part, by enhanced production of an extracellular fibril matrix that binds cells—and their evolutionary interests—together. Though costly to individuals, fibril production greatly enhanced population expansion in groups of interconnected cells. [56]

TWO NATURAL-INSPIRED DECENTRALIZED COOPERATIVE STRATEGIES

In [7] two mechanism of cooperation between individuals inspired in cooperation in natural environment have been analysed: “reciprocity” and “by-product mutualism”. The reciprocity is inspired in blood sharing among vampire bats. It has been observed that female bats in a nest of vampire bats, regurgitate blood meals to other that failed to obtain food in the recent pass. This cooperation was implemented in a way such that agents with a high amount of resources give some amount of it to a closer and weaker agent of the same type. The by-product mutualism is inspired in the behaviour of house sparrows. The sparrows arriving at a patch of food first are more likely to produce chirrup calls. The chirrups call rates are higher when the food resource is divisible and too big to be picked up for a single sparrow. This cooperation was implemented in such a way that when an agent does a beneficial action, like the improvement of a solution, performs a call to the other mutualism agents of the same type that are close in the search space and that then goes to the position of the caller agent. They [same reference] also analyse a model where both type of agents coexist. In one model the two types of cooperating populations co-exist but do not interact between them. In a second model it was allowed cooperation between agents of different type; i.e., using different kind of cooperation. The comparison between the two types of cooperation yields a superiority of the by-product mutualism. The analysis of the co-existence of both types of cooperating individuals, but each one with its own method of cooperation, was not conclusive with respect to the model with just one type of cooperation. When agents using different types of method were allowed to cooperate, their results increase significantly.

ARTIFICIAL IMMUNE SYSTEMS

Artificial Immune Systems (AIS) [12] are based on the communication mechanism of the biological immune systems. In the biological immune system, signal diffusion and dialogue are two kinds of communication schemes available. They take a major role in sharing and passing information during immune response. In *immune diffusion*, the message is passed from one immuno-component to others without any feedback. Another scheme is called *immune dialogue*, where the immune system continuously exchanges molecular signals with its counterparts. Immune sensitivity is determined by context, where self and foreign agents play upon each other. The body is under constant challenge to respond along a continuum of behaviour and needs to adapt accordingly. Signalling is important in biological defence as it allows a cell to move a signal from the outside to the inside, and signalling results in changes to the cell, allowing it to appropriately respond to a stimulus. From an information-processing perspective, the immune system is a remarkable parallel and distributed adaptive system with (partial) decentralized control mechanism. It uses feature extraction, signalling, learning, memory, and associative retrieval to solve recognition and classification tasks. In particular, it learns to recognize relevant patterns, remember patterns that have been seen previously, and use combinatory to construct pattern detectors efficiently. Also, the overall behaviour of the system is an emergent property of many local interactions. These remarkable information-processing abilities of the immune system provide several important aspects in the field of computation. The Artificial Immune nets have been extensively applied to computer security and data mining. Three main mechanisms in the biological immune system have been explored as Artificial Immune Systems (AIS): negative selection, immune network model and clonal selection. A bibliography of AIS recently compiled by Dipankar Dasgupta and Rukhsana Azeem is available from the authors’ webpage [13].

ARTIFICIAL IMMUNE NETWORKS

The Immune Network model made the hypothesis that the immune system maintains an idiotypic network of interconnected B cells for antigen recognition. These cells both stimulate and suppress each other in certain ways that lead to the stabilization of the network. Two B cells are connected if the affinities they share exceed a certain threshold, and the strength of the connection is directly proportional to the affinity they share. In artificial immune network (AIN) models, a B-cell population is made of two sub-populations: the initial population and the cloned population. The initial set is generated from a subset of raw training data to create the B-cell network. The remainders are used as antigen training items. Antigens are then selected randomly from the training set and presented to the areas of the B-cell network. If the binding is successful, then the B-cell is cloned and mutated. The mutation yields a diverse set of antibodies that can be used in the classification procedure. Once a new B cell is created, an attempt is made to integrate it into the network at the closest B Cells. If the new B cell cannot be integrated, it is removed from the population. If no bind is successful, then a B-cell is generated using the antigen as a template and is then incorporated into the network.

CLONAL SELECTION THEORY

The Clonal Selection Principle describes the basic features of an immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigen proliferate, thus being selected against those that do not. The three main features of the Clonal Selection Theory (CST) are that the new cells are copies of their parents (clone) subjected to a mutation mechanism with high rates (somatic hypermutation); elimination of newly differentiated lymphocytes carrying self-reactive receptors; and third proliferation and differentiation on contact of mature cells with antigens. The algorithm (**CLONAL**G) that has been applied ton optimization [48] is based on the clonal selection and affinity maturation principles. It is similar to mutation-based evolutionary algorithms and has several interesting features: 1) population size dynamically adjustable, 2) exploitation and exploration of the search space, 3) location of multiple optima, 4) capability of maintaining local optima solutions, and 5) defined stopping criterion.

NEGATIVE SELECTION ALGORITHMS

The Negative Selection Algorithms has been successfully applied to fault detection. One of the purposes of the immune system is to recognize all cells (or molecules) within the body and categorize those cells as self or non-self. The non-self cells are further categorized in order to induce an appropriate type of defensive mechanism. The immune system learns through evolution to distinguish between foreign antigens (e.g., bacteria, viruses, etc.) and the body’s own cells or molecules. The purpose of negative selection is to provide tolerance for self cells. It deals with the immune system’s ability to detect unknown antigens while not reacting to the self cells. During the generation of T-cells, receptors are made through a pseudo-random genetic rearrangement process. Then, they undergo a censoring process in the thymus, called the negative selection. There, T cells that react against self-proteins are destroyed; thus, only those that do not bind to self-proteins are allowed to leave the thymus. These matured T-cells then circulate throughout the body to perform immunological functions and protect the body against foreign antigens.

For the promising future of this research line we refer to [55].

EVOLUTION COMPUTATION

Evolutionary computation consists of computation methods, often biologically inspired, based on iterative progress, growth or development of a population including guided random search and parallel processing

EVOLUTIONARY ALGORITHM

Evolutionary algorithm (EA) is a generic population-based metaheuristic optimization algorithm. An EA uses some mechanisms inspired by biological evolution: reproduction, mutation, recombination, natural selection and survival of the fittest. Candidate solutions to the optimization problem play the role of individuals in a population, and the cost function determines the environment within which the solutions "live" (see also fitness function). Evolution of the population then takes place after the repeated application of the above operators.

EVOLUTION STRATEGIES

Evolution strategy is an optimization technique based on ideas of adaptation and evolution. Evolution strategies use mutation and selection as search operators. The operators are applied in a loop. An iteration of the loop is called a generation. The sequence of generations is continued until a termination criterion is met.

GENETIC ALGORITHMS

Genetic algorithms [33] can be interpreted like a large ser of solutions (agents) that cooperates in producing better new solutions. Then basic GA represents a cooperation scheme for decentralized cooperation based metaheuristics. In addition, some proposed extensions like cellular Genetic Algorithm and Island models provides additional model for decentralized cooperation in the search for better solutions. The Evolutionary algorithms include several subclasses of methods: Genetic Algorithms (GA), Genetic Programming (GP), Evolutionary Programming (EP) and Evolution Strategies. Tin the so-called structured EA, the populations is decentralized somehow and the subpopulations cooperate during the optimization process. Distributed and cellular models are popular names of strategies that adopt different structures and rules in the communications among the subpopulations.

ADAPTIVE GENETIC ALGORITHM

Adaptive Genetic Algorithm (AGA) comes from the conception of cooperative methods in order to solve multi-objective combinatorial optimization problems. This in of the many cooperation schemes between exact and/or heuristic methods have been proposed in the literature. Adaptive genetic algorithm (AGA) is a new heuristic that is designed for an efficient exploration of the search space. There are several cooperation schemes between AGA and other methods (exact or heuristic). The performance of these schemes has been tested on a bi-objective permutation flow-shop scheduling problem, in order to evaluate the interest of each type of cooperation. [3]

MEMETIC ALGORITHMS

Memetic Algorithms (MA) [38] combine individual local search methods with a population-based global search. They are based on the metaphors of scientific or cultural evolution: phases of individual improvements to ideas or theories alternate with phases of cooperative and competitive interactions on the population level. In the original formulation of ma, a number of individuals (16 in this case) is arranged on a ring. Each individual starts by adopting a random solution and improves it through a local search method such as sa. Individuals then compete with neighbouring individuals or cooperate with distant individuals. Competition results in one individual copying the solution of its neighbour; cooperation results in exchange of information through GA-type crossover operators. The local and global search operations are repeated until a stopping criterion is satisfied.

NEURAL NETWORKS

An artificial neural network [9] is an information or signal processing system composed of a large number of simple processing elements, called artificial neurons or simply nodes, which are interconnected by direct links called connections and which cooperate to perform parallel distributed processing in order to solve a desired computational task. The potential benefits of neural networks extend beyond the high computation rates provided by massive parallelism. The neural network models are specified by the net topology, node characteristics, and training or learning rules. These rules specify an initial set of weights and indicate how weights should be adapted during use to improve performance. Roughly speaking, these computations fall into two categories: natural problems and optimization problems.

MODELS FROM HUMAN COOPERATION

COOPERATIVE TEAM PLAYING

Within living organism there is cooperation at many different levels. There is cooperation in the behaviour of a community of individuals when they work together to achieve some goal using some cooperation mechanism. Human society is probably the largest cooperative system in nature. We use very different mechanisms of cooperation at different levels, at different scopes and at the same time. Team playing is a common cooperative behaviour among human beings where the team members working together through cooperation can solve hard problems which are beyond the capability of any member in the team. Cooperation strategies used in team playing have recently inspired a general optimization method, called cooperative optimization, for attacking difficult optimization problems [36]

To obtain decentralized cooperative heuristic is enough to discover, describe, abstract, and formalize a cooperative system in nature. The cooperative mechanism must be generalized, applied and implemented for the solution of optimization problems. The cooperation model would include the structure for organizing the agents and the rules for the interaction among them; what data are communicated, when it is done and what the agents would do with this information. Because of the richness of the cooperation systems that can be found in nature, it is possible to develop many cooperative metaheuristics inspired by them. The cooperative metaheuristic is decentralized when no special agent in the system dictates others and there is not centralized control to manage the agents. The cooperation mechanisms are defined by the pattern of interactions among the agents and the way of decision making by these agents. Often times a decentralized and self-organized cooperative system in nature have intelligent emerging behaviours through the local and simple interactions of the agents in the system. These smart systems will deal with interesting decentralized cooperative strategies for metaheuristics (see the paper "The Cooperative Optimization Metaheuristic: Inspiration from Nature and Applications" in [37])

Human being, just like ants, beeps and genes, are also part of nature. Intelligent human behaviour also inspire optimization metaheuristic methods. Team playing is a common cooperative behaviour among human beings where its members work together to achieve the best for the team, even if it is not the best for its members. Preference passing and solution compromising are two key strategies used in team playing. Preference passing allows each team member to act according to the preferences of other team member on the solution process. Solution compromising enables the team members to resolve possible conflicts among their preferences at choosing the best solution from their joint point of view. [35]

DECENTRALIZED OPTIMIZATION BY NASH BARGAINING

Decentralized Optimization via Nash Bargaining [58] is a new decentralized optimization method for solving multi-player coordination problems using. The algorithm utilizes the Nash Bargaining solution as the preferable outcome for all players among the set of Pareto optimal points, under assumptions of convexity. It is based on the concept on a multi-agent kinematic trajectory planning problem with collision avoidance. The analysis and numeric comparison of complexity between centralized and decentralized penalty method based optimization suggest operation regimes where the decentralized method incurs no increase in complexity and even improvement in computation time proportional to the number of players over the centralized method.

CO-EVOLUTION MODELS

Evolutionary optimization models generally utilize a static selection function; individuals are selected on the basis of how well they comply to the objective function. In a coevolutionary model the success of individuals (i.e. their number of offspring) depends on circumstances that represent only a subset of the objective function. The resulting evolutionary dynamics may lead to continuous evolutionary change, i.e. red queen dynamics, or speciation of the populations, or to the evolution of general behaviour, i.e. optimization of individual behaviour. The sparseness of the fitness evaluation, i.e. individuals are only evaluated on a subset of all fitness cases, may have as side-effects an increased efficiency in the optimization process, it gives more freedom to evolve with respect to the complete objective funtion, and it may lead to a continuous effective selection gradient. Several co-evolution approaches are related with cooperative strategies and optimization. [32]

EXTREMAL OPTIMIZATION

Extremal Optimization (EO) [6] is an optimization heuristic inspired by the Bak-Sneppen model of self-organized criticality from the field of statistical physics. The Bak-Sneppen model [3] is a simple model of co-evolution between interacting species. It was developed to show how self-organized criticality may explain key features of the fossil record, such as the distribution of sizes of extinction events and the phenomenon of punctuated equilibriums. The model dynamics repeatedly eliminates the least adapted species and mutates it and its neighbours to recreate the interaction between species. Self-organized criticality (SOC) is a property of dynamical systems which have a critical point as an attractor. These are non-equilibrium systems that evolve through avalanches of change and dissipations that reach up to the highest scales of the system. SOC is said to govern the dynamics behind some natural systems that have these burst-like phenomena including landscape formation, earthquakes, evolution, and the granular dynamics of rice and sand piles. It has been observed that self-similar ("critical") structures are quite ubiquitous in nature, and the question as to the dynamical origin of those structures arises. Some years ago, Self-Organized Criticality (SOC) was proposed as one mechanism to describe the basic mechanism that creates generic scale-free behaviour. According to SOC, such behaviour emerges when an externally driven, dissipative system organizes itself into a state in which all spatial and temporal events are correlated over many orders of magnitude.

The Extremal Optimization method is inspired by the far-from-equilibrium dynamics often found in nature where extremely bad adapted components of a system are selectively driven to extinction, leaving behind complex and highly adapted structures (such as eco-systems). Extremal Optimization evolves a single solution and makes local modifications to the worst components. This requires that a suitable representation be selected which permits individual solution components to be assigned a quality measure ("fitness"). This differs from holistic approaches such as ant colony optimization and genetic algorithm that assign equal-fitness to all components of a solution based upon their collective evaluation against an objective function. The algorithm is initialized with an initial solution, which can be constructed randomly, or derived from another search process.

The technique is a fine-grained search, and superficially resembles a hill climbing (local search) technique. A more detailed examination reveals some interesting principles, which may have applicability and even some similarity to broader population-based approaches (evolutionary computation and artificial immune system). The governing principle behind this algorithm is that of improvement through selectively removing low-quality components and replacing them with a randomly selected component. This is obviously at odds with genetic algorithms, the quintessential evolutionary computation algorithm that selects good solutions in an attempt to make better solutions.

The resulting dynamics of this simple principle is firstly a robust hill climbing search behaviour, and secondly a diversity mechanism that resembles that of multiple-restart search. Graphing holistic solution quality over time (algorithm iterations) shows periods of improvement followed by quality crashes (avalanche) very much in the manner as described by punctuated equilibrium. It is these crashes or dramatic jumps in the search space that permit the algorithm to escape local optima and differentiate this approach from other local search procedures. Although such punctuated-equilibrium behaviour can be "designed" or "hard-coded", it should be stressed that this is an emergent effect of the negative-component-selection principle fundamental to the algorithm.

The Extremal Optimization metaheuristics [54] has been applied to several Logistics problems, among them one of the first paper on EO deal with the travelling salesman problem.

NATURAL COMPUTING

MEMBRANE COMPUTING

Membrane Computing (MC) includes distributed computing models inspired in the cooperation of cells in tissues, organs and organisms. The computational devices of MC are the membrane systems which are inspired by the structures and functioning of living cells. There are basically two ways of consider these computational devices: cell-like membrane systems, using membranes arranged hierarchically, inspired from the structure of the cell, and tissue-like membrane systems using membranes placed in the nodes of a graph, inspired from the cell intercommunication in tissues. In [34] cell-like and tissue-like Membrane Systems as Recognizer devices are presented.

Membrane computing [48] aims at defining computational models which abstract from the functioning and structure of the cell. Unlike bacterium, which generally consists of a single intracellular compartment, an eucaryotic cell is sub-divided into functionally distinct compartments through membranes. A class of computing devices called *membrane systems*, or P systems, are defined to have three essential features: a *membrane structure* consisting of a hierarchical arrangement of several compartments inside a main membrane (the skin) defined as regions delimited by membranes; *regions* defined by membranes have objects corresponding to chemical substances and can be described by string of symbols; and *rules* assigned to the regions of the membrane structure, acting upon the objects inside that evolves according to them (rules encode processes for producing, eliminating or moving objects. A *configuration* consists of a membrane structure and a family of multi sets associated with each region of the structure. A configuration is transformed to another configuration by applying evolution rules (in a non-deterministic way) to the object inside the region in a decentralized manner. From an initial configuration, a *computation* consists in applying these transformations. The computation halt in a configuration where no more rules can be applied and the result is given by the multiset corresponding to the skin membrane that communicates with the environment.

A membrane system can include *active membranes*, where membrane division can be applied in the evolution of the systems. In these models, an object in a region can be introduced into a membrane in that region or can be sent out through a membrane to a region immediately outside its region. A membrane can be dissolved or can be divided between two membranes. More complicated division rules could be considered. In these cases, the objects in the region involved in the division can change. In the tissue-like membrane systems, the cells can communicate following a graph and can also be divided in a similar way that membrane do in a cell-like membrane systems. Membrane systems have been applied to several optimization problems, some of them related to Logistics, like knapsack and bin-packing problem [10]. The relationship between P systems and decentralized computation is considered in [5]

LOGISTICS PROBLEM.

The optimisation problems found in Logistic are strategic or operative decisions concerning loading, location and routing. Most of the real problems are mixed problems involving these three types of decision. However, in the other size, the standard problems on these types belong to the NP-complete class and have received a great interest in the specialized literature. These problems are, respectively, the Bin Packing Problem (SPP), the p-Median Problem (PMP) and the Travelling Salesman Problem (TSP). They and their extensions or hybridisation are in the most appealing combinatorial optimization problems with application in Logistics. These problems are usually one of the first problems considered in the research on any new meteheuristic, and in particular, in the nature-inspired decentralized cooperative metaheuristic strategies described above. Some of the early references on those strategies for some of these problems are shown in the following table

	Loading	Location	Routing
	BPP	p-Median	TSP
Ant Colony Ooptimization	[25]	[42]	[20]
Particle Swarm Optimization	[43]	[53]	[52]
Artificial Inmune Systems	[45]	[46]	[15]
Genetic Algorithm	[27]	[41]	[44]
Memetic Algorithm	[49]		[47]
Artificial Neural Networks	[2]	[14]	[31]

CONCLUSIONS

The social behaviour found in all the levels in the natural world are good sources of inspiration for the design of decentralized cooperation strategies for Metaheuristics. Some of these strategies have already been analyzed in the specialized literature; however there is room for new developments. The adaptability to the environment of these systems is also a very good inspiration for the design of optimization procedure in changing circumstances. If the instances data varies slightly the nature-inspired agents and systems adapts to the changes in real time. These strategies give rules to know when and how interchange the information between the cooperating agents; however, the selection of this information is relevant in the success of the application. The knowledge on the application fields, like Logistics, can conduct this selection. However the use of Data Mining technique will be relevant for this task.

REFERENCES

[1] *Handbook of Collective Intelligence*, http://www.socialtext.net/mit-cci-hci/index.cgi

[2] A. Bahrami, C.H. Dagli. (1994) “Hybrid Intelligent Packing System (HIPS) through integration of Artificial Neural Networks, Artificial Intelligence, and mathematical programming”. Applied Intelligence, Vol. 4(4):321-336, 1994

[3] P. Bak, K. Sneppen (1993), "Punctuated equilibrium and criticality in a simple model of evolution." Phys. Rev. Lett. 71, 4083–4086

[4] M. Basseur (2006). “Design of cooperative algorithms for multi-objective optimization: application to the flow-shop scheduling problem”. 4OR: A Quarterly Journal of Operations Research. Volume 4, No. 3, 2006..

[5] F. Bernardini, F.J. Romero, M. Gheorghe, M.J. Pérez-Jiménez. (2006) “A modelling approach based on P systems with bounded parallelism”. Lecture Notes in Computer Science, in press.

[6] S. Boettcher and A. Percus (2000). “Nature's way of optimizing”. Artificial Intelligence, Vol. 119, 2000, pp. 275-286

[7] F. Bonachela, D. Pelta, A. Sancho-Royo (2006). “Analysis and Comparisons of Two Nature-Inspired Cooperative Strategies”, NICSO 2006, Granada

[8] E. Bonabeau, M. Dorigo, G. Theraulaz. “Swarm Intelligence: From Natural to Artificial Systems”. Oxford University Press, 1999.

[9] A. Cichocki, R. Unbehauen. “Neural Networks for Optimization and Signal Processing”. John Wiley 1993.

[10] G. Ciobanu, Gh. Paun, M. J. Pérez-Jiménez (eds.). “Applications of Membrane Computing. Natural Computing” Series, Springer-Verlag, 2006.

[11] M. Clerc. “Particle Swarm Optimization”. ISTE, 2006.

[12] D. Dasgupta (2006). “Advances in Artificial Immune Systems”. To appear in IEEE Computational Intelligence Magazine.

[13] D. Dasgupta, R. Azeem (2006). “Artificial Immune Systems: A Bibliography”. Computer Science Department. The University of Memphis, USA. CS Technical Teport No. CS-06-003, 2006.

[14] E. Domínguez Merino, J. Muñoz Pérez and J. Jerez Aragonés. “Neural Network Algorithms for the p-Median Problem” ESANN'2003 proceedings. 23-25 April 2003, pp. 385-391, Bruges (Belgium),

[15] de Castro, L. N. & Von Zuben, F. J. (2000), "Artificial Immune Systems: Part II – A Survey of Applications", Technical Report – RT DCA 02/00, p. 65.

[16] L.N. de Castro, F.J. Von Zuben. “Recent Developments in Biologically Inspired Computing”. Idea Group Publishing, 2005

[17] K. De Meyer (2004). “Foundations of Stochastic Diffusion Search”. Ph.D. Thesis. Department of Cybernetics. The University of Reading (UK) 2004.

[18] F. Denison, T. Kiers, P. Bailey. “Host sanctions and the legume-rhizobium mutualism”. Nature, 425:78-81, 2003.

[19] M. Dorigo, C. Blum. “Ant colony optimization theory: A survey”. Theoretical Computer Science, 344(2-3):243-278, 2005.

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

[21] M. Dorigo, V. Maniezzo, A. Colorni, “Positive Feedback as a Search Strategy”. Technical Report No. 91-016, Politecnico di Milano, Italy, 1991.

[22] M. Dorigo, V. Maniezzo, A. Colorni. “Optimization by a colony of cooperating agents”, IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26(1):29-41, 1996.

[23] M. Dorigo, G. Di Caro, “The Ant Colony Optimization Meta-Heuristic”. In D. Corne, M. Dorigo and F. Glover, editors, New Ideas in Optimization, McGraw-Hill, 11-32, 1999.

[24] M. Dorigo, T. Stützle. “Ant Colony Optimization”. MIT Press, 2004.

[25] Ducatelle, F. (2001). “Ant colony optimisation for bin packing and cutting stock problems”. Master's thesis, University of Edinburgh.

[26] A. P. Engelbrecht. “Fundamentals of Computational Swarm Intelligence”. Wiley, 2005.

[27]E. Falkenauer. "A hybrid genetic algorithm for bin packing". Journal of Heuristics, 2(1):5–30, 1996.

[28]P.G. Flikkema, J.G. Leid. "Bacterial communities: a microbiological model for swarm intelligence". Proceedings of the IEEE Swarm Intelligence Symposium, 2005. pp. 416- 419

[29]N. Forbes. "Imitation of life: How Biology is Inspiring Computing". MIT Press, 2005

[30]N.R. Franks, T. Richardson (2006). "Teaching in tandem-running ants". Nature, 439:153-153.

[31]B. Fritzke and P. Wilke. "FLEXMAP. A neural network with linear time and space complexity for the traveling salesman problem", Proc. IJCNN-90, Singapore, pp. 929--934, 1991.

[32]N. García-Pedrajas, E. Sanz-Tapia, D. Ortiz-Boyer, C. Hervás-Martínez (eds.) "Introducing Multi-objective Optimization in Cooperative Coevolution of Neural Networks". LNCS 2084/2001. Springer

[33]D.E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning". Addison-Wesley, 1989.

[34]M.A. Gutiérrez-Naranjo, M.J. Pérez-Jiménez, A. Riscos-Núñez, F.J. Romero Campero, "Cell-like and Tissue-like Membrane Systems as Recognizer Devices". NICSO, 2006, Granada (SPAIN).

[35]X. Huang (2004) "Cooperative optimization for solving large scale combinatorial problems". In Theory and Algorithms for Cooperative Systems, D. Grundel, R. Murphey, P.M Pardalos (eds.) Series on Computers and Operations Research - Vol. 4, 2004 World Scientific, pp. 117-156

[36]X. Huang (2006), "From Cooperative Team Playing to an Optimization Method", NICSO, 2006. Granada (SPAIN).

[37]X. Huang (2006). "The Cooperative Optimization Metaheuristic: Inspiration from Nature and Applications". International Conference on Intelligent Computing. August 16–19, 2006 Kunming, China. Special Session on Intelligence Computation and Its Application. Lecture Notes in Computer Science. Volume 4114/2006. Pages 1246-1251

[38]J. Kennedy, R.C. Eberhart. "Particle swarm optimization". Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ. pp. 1942-1948, 1995

[39]W.E. Hart, N. Krasnogor, J.E. Smith, (eds.). "Recent Advances in Memetic Algorithms", Springer, 2004.

[40]W.E. Hart, N. Krasnogor, J.E. Smith. "Memetic Evolutionary Algorithms", in Recent Advances in Memetic Algorithms, Springer, Springer, 2004.

[41]J. Kratica, D. Tosic, V. Filipovic, I. Ljubic "Solving The Simple Plant Location Problem By Genetic Algorithm" RAIRO Operations Research, 35:127-142, 2001

[42]T. V. Levanova. "The Ant Colony Optimization Algorithm for some Discrete Location Problems". 12th International Conference of the EWM Association. Volgograd, Russia; 18--24 September 2005.

[43]J.-H. Lin, C.-K. Wang, and C.-H. Lee. "Particle Swarm Optimization for Web Newspaper Layout Problem". The IASTED International Conference on Web Technologies, Applications and Services (WTAS 2006), Calgary, Alberta, Canada, 2006

[44]P. Merz, B. Freisleben. "Genetic Local Search for the TSP": New Results, in Proceedings of the 1997 IEEE International Conference on Evolutionary Computation, IEEE Press, pp. 159-164, Indianapolis, USA, 1997.]

[45]K. Mori, M. Tsukiyarna and T. Fukuda. "Load Dispatching Planning by Immune Algorithm with Diversity and Learning". 7th Int. Conf. on Systems Research, Informatics and Cybernetics, Vol.11, pp.136-141, 1994.

[46]K. Mori, M. Tsukiyama and T. Fukuda. "Immune Algorithm with Searching Diversity and its Application to Resource Allocation Problem". The Trans. of the Institute of Electrical Engineers of Japan, Vol.113-C, No.10, pp.872-878 (in Japanese), 1993.

[47]P. Moscato and F. Tinetti. “Blending Heuristics with a Population-Based Approach: A Memetic Algorithm for the Traveling Salesman Problem”, Report 92-12, Universidad Nacional de La Plata, 1992.

[48]Gh. Paun, (2002). “Membrane Computing. An Introduction”, Springer, Berlin.

[49]C.R. Reeves, “Hybrid Genetic Algorithms for Bin-packing and Related Problems”, Annals of Operations Research, 63 (1996), pp. 371-396.

[50]M. Resnick “Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds”. MIT Press, 1997.

[51]J. Riget, J. S. Vesterstrøm. “A diversity-guided particle swarm optimizer – the arPSO”. Technical report, EVALife, Dept. of Computer Science, University of Aarhus, Denmark, 2002.

[52]B. R. Secrest, G.B. Lamont. “Communication in particle swarm optimization illustrated by the traveling salesman problem”. Proceedings of the Workshop on Particle Swarm Optimization 2001, Indianapolis, IN. 2001

[53]Mehmet Sevkli, Ali Riza Guner. “A New Approach to Solve Uncapacitated Facility Location Problems by Particle Swarm Optimization”. Proceedings of 5th International Symposium on Intelligent Manufacturing Systems, May 29-31, 2006: pp. 237-246

[54]F.L. de Sousa, F.M. Ramos, R.L. Galski, I. Muraoka. “Generalized Extremal Optimization: A new meta-heuristic inspired by a model of natural evolution”. Chapter 3 in L.N. de Castro, F.J. Von Zuben (eds.) Recent Developments in Biologically Inspired Computing. Idea Group Publishing, 2005

[55]J. Timmis (2006). “Artificial Immune Systems - Today and Tomorrow”. To appear in Special Issue on Natural and Artificial Immune Systems. *Natural Computation.*

[56]G. J. Velicer, Y.N. Yu. (2003) "Evolution of novel cooperative swarming in the bacterium Myxococcus xanthus" NATURE. VOL 425. 2003

[57]X. Wang, X.Z. Gao, S.J. Ovaska (2004) “Artificial immune optimization methods and applications: A survey”. Proceedings of the International Conference on Systems, Man and Cybernetics, The Hague , The Netherlands, 2004, pp. 3414-3420

[58]S.L. Waslander, G. Inalhan, C.J. Tomlin (2004) “Decentralized Optimization via Nash Bargaining”, in D. Grundel, R. Murphey P.M. Pardalos (eds.) *Theory and Algorithms for Cooperative Systems*. Series on Computers and Operations Research - Vol. 4., World Scientific 2004.