MEMETIC ALGORITHMS

Minh Nghia Le
Nanyang TechnologicalUniversity, Singapore.

Ferrante Neri
De Montfort University, UK

Yew Soon Ong
Nanyang Technological University, Singapore

Keywords: memetic computing, memetic algorithms, evolutionary algorithms, computational intelligence, meta-heuristics, adaptation, symbiosis, algorithmic design

Contents

Summary

The field of computational intelligence (CI) has taken flight and for the last decades served in a large part of computer science and engineering literature as a field that devotes to the development and implementations of various new methodologies for solving complex problems successfully. Memetic Algorithm (MA), also commonly known as hybrid evolutionary algorithms (EAs), or genetic local search, represents a recent established field of CI that has attracted increasing research interest. In parallel to the MA definition and early diffusion, a strictly related concept, i.e. hyper-heuristic, was defined as an algorithm composed of multiple algorithmic components coordinated by a supervisor element. Recent developments of memetic computing can lead to the fusion of (canonical) memetic algorithms and meta-heuristics, especially of the adaptive rules in the coordination mechanisms. Since MAs were not proposed as specific optimization algorithms, but as a broad class of algorithms inspired by the diffusion of the ideas and composed of multiple existing operators, the community started showing an increasing attention towards these algorithmic structures as a general guideline for addressing specific problems. In this chapter, our focus is on the design of memetic frameworks for solving continuous complex optimization problems. Some key factors responsible for the success of these frameworks are identified and presented into two levels of design as guideline for the practitioners.

1. Introduction

Optimization is a classical problem that arises in various domains ranging from physics, biology, engineering designs to a plethora of other real world applications found in our everyday life. The design of complex systems encompasses a wide range of activities whose goal is nonetheless to find the optimum characteristics of a product before it is manufactured. The complexity of today's problems can be attributed to challenges involving non-linearity, high multi-modality, uncertainty and computationally expensive problems and other real-time demands. In such scenarios, the use of conventional methods is often deemed as ineffective or inadequate, in general, mainly due to the lack of sufficient *prior* knowledge (hypotheses) available on the problem to solve. For example, the analytical expression of the merit function(s) to minimize, often also known as the cost or objective or fitness function in evolutionary computation, are often unavailable. Further, it is becoming a common practice that the objective function(s) manifest in the form of computational simulations or through physical experiments/measurements. Under such conditions, which is typical of today's real-world complex problems, the optimization must be conducted by treating the objective functions as "black boxes" to optimize. For these reasons, the field of computational intelligence (CI) has taken flight and for the last decades served in a large part of computer science and engineering literature as a field that devotes to the development and implementations of various new methodologies for solving complex problems successfully.

The field of optimization is a study that has been embraced extensively by researchers from various disciplines, with many algorithms and implementations that are now made available and used in the different communities. Early well established meta-heuristics and CI approaches include Simulated Annealing, Evolutionary and Swarm Intelligence. While separate paradigms have been independently developed in parallel tracks, a part of the computer science and engineering community has realized that proper combination of nature-inspired and culture-inspired operators can lead to the generation of efficient optimizers which may outperform, by several orders of magnitude, each of its stand-alone components. Among the early attempts to demonstrate this was reported in the 1980's in the name of hybrid evolutionary algorithms or memetic algorithms, see (Neri et al, 2012; Ong et al, 2010; Chen et al 2011; Goh et al 2009) for some excellent expositions of the field.

Memetic Algorithm (MA), also commonly known as hybrid EAs, Baldwinian EAs, Lamarckian EAs, or genetic local search, represents a recent established field of CI that has attracted increasing research interest where a growing number of publications appearing in a plethora of international journals and conference proceedings has been noted. The earliest form of Memetic Algorithms (Goldberg, 1989; Moscato, 1989; Moscato 1999; Smith et al. 2009) was first introduced as a marriage between population-based global search and individual learning, where the latter is also often referred to as a local search or meme, capable of refinement or learning. Fundamentally rooted on Darwinian principles of natural evolution and Dawkins notion of a meme, many modern evolutionary algorithms in the field of computational intelligence have been designed and crafted specifically for addressing particular problems or domains, and with significant success reported (Ishibuchi et al. 2003; Krasnogor et al. 2002; Ong at al. 2003; Ong at al. 2006; Tang et al. 2009, Tirronen at al., 2008). Memetic algorithms have been used successfully to solve a wide variety of engineering design problems and often shown to generate higher quality solutions more efficiently than canonical evolutionary algorithms (Hart 1994; Ong et al. 2004; Lewis et al. 2007; Wang et al. 2010). A discussion on the different depictions of MAs inspired from Dawkins's theory of Universal Darwinism is provided in (Nguyen et al. 2008).

In parallel to the MA definition and early diffusion, a strictly related concept, i.e. hyper-heuristic, was defined. A hyper-heuristic is an algorithm composed of multiple algorithmic components coordinated by a supervisor element, this element can be a heuristic itself, or, in modern implementations, a machine learning technique. Both, hyper-heuristics and MAs are thus optimizers composed of multiple search operators to perform the search. Although both the algorithms are heterogeneous structures, the characterization of a MA is generally about the algorithmic components building while the characterization of hyper-heuristics is on the coordination rule of the components. Recent developments of memetic computing can lead to the fusion of (canonical) memetic algorithms and meta-heuristics, especially of the adaptive rules in the coordination mechanisms. As such, the term "memetic algorithm" shall be used to represent these fields of research throughout this chapter.

The importance and diffusion of MAs should be put into relationship with the No Free Lunch Theorem (NFLT), see (Wolpert at al. 1997). The NFLT proves that the average performance of any pair of algorithms A and B across all possible problems is identical. Thus, if an algorithm performs well on a certain class of problems, then it necessarily pays for that with degraded performance on the set of all remaining problems, as this is the only way that all algorithms can have the same performance averaged over all functions. Strictly speaking, the proof of NFLT is made under the hypothesis that both the algorithms A and B are non-revisiting, i.e., the algorithms do not perform the fitness evaluation of the same candidate solution more often than once during the optimization run. Although this hypothesis is *de facto* not respected for most of the computational intelligence optimization algorithms, the concept that there is no universal optimizer had a significant impact on the scientific community. For decades, researchers in optimization attempted to design algorithms having a superior performance with respect to all the other algorithms present in literature. This approach is visible in many famous texts published in those years, e.g., (Goldberg 1989). After the NFLT diffusion, researchers in optimization had to dramatically change their view about the subject. More specifically, it has become important to understand the relationship between the components of the proposed algorithm A and a given optimization problem f. Thus, the problem f became the starting point for building up a suitable algorithm. The optimization algorithm needs to specifically address the features of problem f.

Since MAs were not proposed as specific optimization algorithms, but as a broad class of algorithms inspired by the diffusion of the ideas and composed of multiple existing operators, the community started showing an increasing attention towards these algorithmic structures as a general guideline for addressing specific problems.

In this chapter, our focus is on the design of memetic frameworks for solving continuous complex optimization problems. Some key factors responsible for the success of these frameworks are identified and presented into two levels of design as guideline for the practitioners. It is well established that the main purpose of designing a successful MA hybrid search lies in balancing well between generality (through stochastic variations) and problem specificity (through individual learning) (Hart et al. 2004; Moscato 1999; Nguyen et al. 2008, Paenke et al. 2009; Renders et al. 1994). As such, the micro-level design of memetic framework described in Section 2 discusses several important algorithmic configurations responsible for such balance, including the choice of learning mode, the learning frequency, the learning intensity in term of computational budget and others. On the other hand, the macro-level design focuses more on the algorithmic component aspects of the framework, i.e., stochastic variation and individual learning operators. In particular, representative memetic operators are reviewed and discussed in Sections 3.1 and 3.2. Besides, several state-of-the-art coordination mechanisms of memetic operators are also included in Section 3.3 as some recent advancing developments of memetic algorithms. Last but not least, Section 4 concludes the chapter and outlines some potential notable future research directions of memetic computation.

2. Micro-level Design of Memetic Framework

An outline of the basic Memetic Algorithm composed of the stochastic variation operators and individual learning to refine the offspring solutions is given in Algorithm 2. In particular, stochastic variation operators, such as crossover and mutation in genetic algorithm, present as components of a population-based (or global search) algorithm. On the other hand, individual-based search operator, also known better as local search, individual learning or lifetime learning, involves the process of searching for an improved solution (if it exists), starting from a given vector of decision variables (Bunday 1995).

In the first step, a population of individuals is initialized either randomly or using design of experiment techniques such as Latin hypercube sampling. The evaluated population of individuals then undergoes natural selection, for instance, via fitness-proportional or tournament selection. In Algorithm 2, the selection and replacement schemes emulate the effects of "the survival of the fittest" in natural selection. Replacement methods are similar to parent selection operators that determine which individuals shall survive across the generations. A great number of selection operators have been proposed in the literature (Back et al. 1997; Goldberg et al. 1991), extending from fitness proportional and stochastic universal selections (Baker 1987) to tournament selection (Brindle 1981) and Boltzmann selection (Cai et al. 2006). The choice of selection operator would largely depend on the selection pressure desired in the search

Each individual $\mathbf{x}$ in the reproduction pool is evolved to arrive at offspring $\mathbf{y}$ using stochastic variation operators such as crossover and mutation. The criteria for offspring $\mathbf{y}$ or the subset of individuals Ω_{il} that undergo individual learning are defined by the selection schemes (i.e., random sampling, stratified sampling or elitism) and/or the frequency of individual learning parameter f_{il}, where the latter determines how often individual learning is used in the population per generation. Individual learning $L(y)$ is applied on the selected offspring $\mathbf{y}$ with a computational budget of C_{il} to arrive at the refined solution $\mathbf{z}$. The parent population are then replaced by the offspring to form a new population and the entire process repeats until the specified stopping criteria is satisfied.

Algorithm 1 Memetic Algorithm
1: Generate an initial population
2: **while** Stopping conditions are not satisfied **do**
3: Evaluate all individuals in the population
4: Select individuals for the parents pool $\mathbf{P}^t$ via selection operator $S(.)$
5: **for** each individual x in **P do**
6: *Evolve* x to offspring y according to stochastic variation operators
7: **if** y selected to undergo individual learning **then**
8: Refine y to z through individual learning operator L(y) within the computational budget C_{il}
9: Proceed in the spirit of Lamarckian or Baldwinian learning
10: **end if**
11: Replace offspring into the population
12: **end for**
13: **end while**

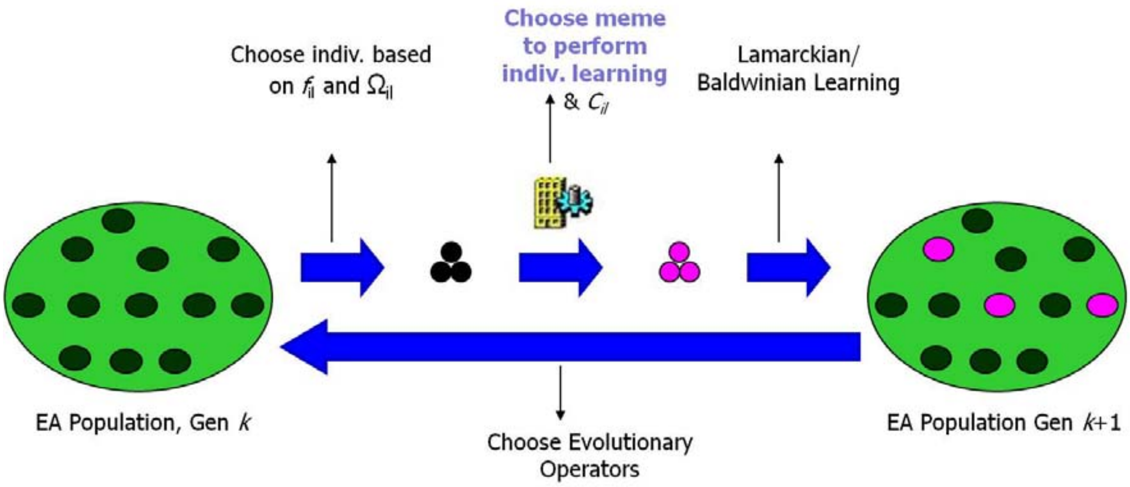

Figure 1. Micro-level Design of Memetic Algorithm.

Based on the canonical MA presented in Algorithm 2, the next subsections shall discuss some important algorithmic configurations responsible for the balance between global search and local searches (Hart et al. 2004; Nguyen et al 2008; Nguyen et al. 2009;, Ong et al. 2010; Chen et al. 2011; Neri et al. 2012), defined in this chapter as the micro-level design of memetic framework and depicted in Fig. 1.

2.1. Modes of Learning

It is worth noting that individual learning can be incorporated in memetic algorithm as a form of population initialization, i.e., *before* the population-based search, to enhance the search performance as contrast to the typically-used simple random population initialization scheme. For *interleaved* hybrid procedures, on the other hand, individual learning is conducted after undergoing the stochastic variation or reproduction operator(s). In other hybrids, refinement is incorporated *after* the population-based search as a form of post-processing to fine-tune or improve the precision of the solution found by the EA. From the literature, the *interleaved* hybrid procedures are the most common and popular configuration used in MA, as outlined in Algorithm 2.

Next, let us consider a classical MA composing of an evolutionary framework and an individual learning phase (local search) that periodically selects an individual from the population with the attempt to enhance it. When the output of the local search, i.e., the improved solution, is produced a natural question arises: what to do with the improved solution and how to pass this new piece of information to the population of candidate solutions, while even facilitating its replications across generations?

In the literature, two basic modes of individual learning (or inheritance schemes) are often discussed, namely, *Lamarckian* and *Baldwinian* learning (Ong et al. 2006) (line 8 of Algorithm 2). Lamarckian learning forces the genotype to reflect the result of improvement in individual learning by placing the locally improved individual back into the population to compete for reproductive opportunities (Houck et al. 1996; Krasnogor 2002; Ong et al. 2004). In diverse contexts, Lamarckian memetic algorithms have also been used under the name of hybrid evolutionary algorithm, Lamarckian evolutionary algorithm, or genetic local search. Baldwinian learning, on the other hand, only alters the fitness of the individuals and the improved genotype is not encoded back into the population. Let $\mathbf{x}$ and $\mathbf{x}^{\text{imp}}$ denote the initial and improved solutions after undergoing refinement. Algorithmically, Lamarckian learning returns $\left(\mathbf{x}^{\text{imp}}, f\left(\mathbf{x}^{\text{imp}}\right)\right)$ to the population while Baldwinian learning return $\left(\mathbf{x}, f\left(\mathbf{x}^{\text{imp}}\right)\right)$ instead.

Although Lamarck's theory of evolution has generated controversies and doubts from biology, the potentials and contributions of Lamarckian learning in computational evolutionary systems have been significant (Jablonka et al. 1995; Ho 1996). It is worth emphasizing that most successful MAs to date are designed in the spirit of Lamarckian learning which exhibits clear advantage on problems in non-changing environments (Merz 2000; Merz 2004; Whitley et al. 1994). On the other hand, Baldwinian learning is thought as a mechanism that does not disturb the evolution of the solutions nor impedes the diversity of the population. As such, Baldwinian learning is deemed as more appropriate for problems in dynamic or uncertain environments (Ong et al. 2006; Plaenke et al. 2007; Sasaki et al. 1997; Sendhoff et al. 1999). A comparative study, for instance, has also been conducted in (Whitley et al. 1994).

2.2. Algorithmic Parameters

From a survey of the field (Krasnogor et al. 2005), the basic configuration of a memetic algorithm can be summarized (but not limited to) by three core parameters

- The selection scheme for constructing the subset of individuals Ω_{il} that should undergo individual learning, such as random sampling, stratified sampling or elitism (Nguyen et al. 2007).
- Frequency of individual learning f_{il}, which defines how often individual learning is applied on the population throughout the search or the proportion of the population that will undergo individual learning in each generation.

- The maximum computational budget or learning intensity C_{il} allocated for the individual learning phase defines how long each learning process should proceed for. A larger value of C_{il} gives more computational budget or greater emphasis on improving each individual chromosomes, thus leading to higher level of convergence or accuracy in the solution quality.

One of the conventional topics pertinent to the MA hybrid design is to identify *which individuals of the search population should undergo individual learning*, where for instance fitness and distribution-based strategies have been proposed by (Land 1998) and (Nguyen et al. 2009). It is worth to highlight the empirical study in (Nguyen et al. 2007) which showed that the choice of selection schemes in MA largely depends on the characteristics of objective function with less impact by the individual learning intensities.

On the question pertaining to how often individual learning should be used, the effect of individual learning on MA search was investigated in (Hart 1994) where various configurations at different stages of the search were considered. As an empirical guideline, (Nguyen et al. 2007) noted that it may be appropriate to undergo individual learning on half of the MA population while highlighted that under some given fixed computational budget, a good balance between C_{il} and f_{il} is necessary to ensure superior search performance in the MA. In this direction, (Ku et al. 200) also suggested to apply learning on every individual when the computational complexity of the learning procedure is low. Schemes to adapt the *frequency of individual learning* based on search diversity and fitness distribution criteria have also been considered by Molina *et al.* in (Lozano et al. 2004; Molina et al. 2008).

To address the overall balance of stochastic variation and individual learning in search, a theoretical upper bound on *the computational budget* to allocate was proposed in (Nguyen at al. 2009). The bound provided the means to adapt various design issues of MA simultaneously, and at runtime, from which individuals that should undergo individual learning, to the amount of computational budget allocated for learning. In addition, the concept of local search chains to adapt the intensity of individual learning was also introduced in (Molina et al. 2010). To alleviate the potentially high intensity and computational budget incurred in individual learning, especially when dealing with real world complex problems plagued with computationally expensive objective functions, management schemes to adapt the use of approximation models or surrogates in lieu of the original objective functions (Jin 2005; Lim et al. 2010) were also considered.

3. Macro-level Design of Memetic Framework

In memetic algorithms, as represented in Fig. 2, researchers have been exploring on various hybridizations of search operators towards the development and manual crafting of specialized algorithms that solve a specific problem or a set of problems effectively. The success of memetic algorithm is thus often very much reliance on the degree of domain knowledge the human expertise holds.

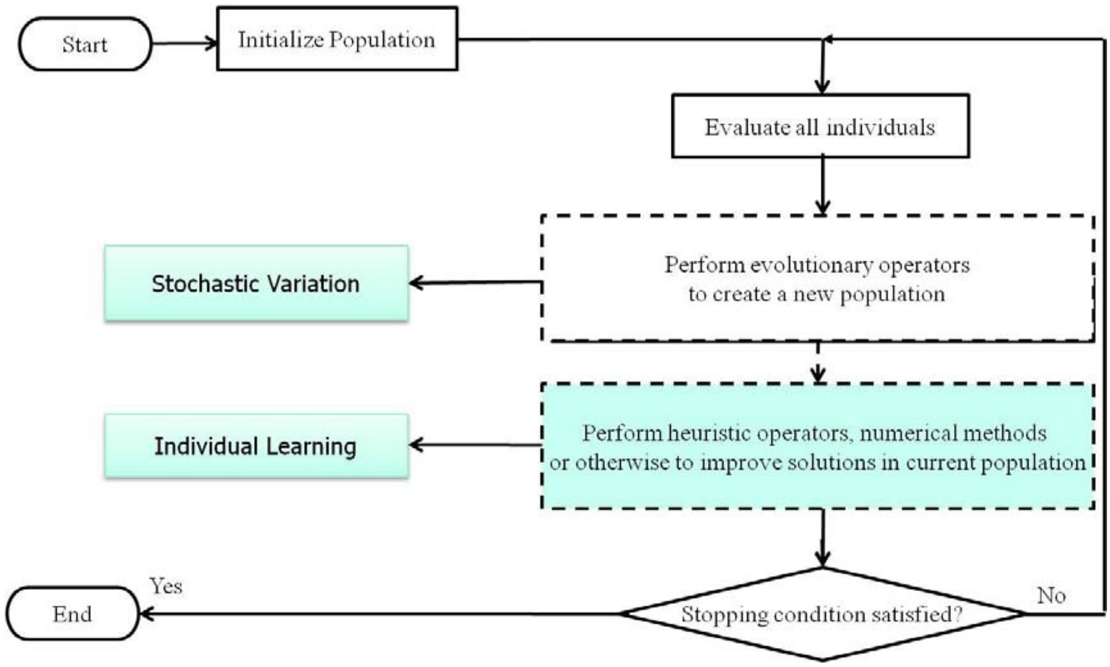

Figure 2. Macro-level Design of Memetic Algorithm.

For instance, the hybridizations of genetic operators with individual-based search methods have manifested as hybrid real-coded Genetic Algorithm with female and male differentiation (RCGA-FMD)(Garcia-Martinez et al. 2005), approximate probabilistic memetic framework based on GA-DSCG (APrMF) (Nguyen at al. 2009), and memetic algorithm with local search chaining (MA-LSCh-CMA) (Molina et al. 2008). A review of different hybridizations of genetic algorithm with diverse individual learning strategies that employ gradient information is reported in (Li et al 2008). On the other hand, accelerating differential evolution using an adaptive local search (DEahcSPX) (Noman et al. 2008) represents an example of combining DE's stochastic operators with local search to accelerate the search progress. Particle Swarm CMA-ES (Muller et al. 2009) denotes an example of the hybrid MA in which CMA-ES is employed as the individual learning procedure with the PSO population-based search. Another notable example is the estimation of distribution algorithm (EDA) with an ant-miner local search proposed in (Aickelin et al. 2006) for solving the nurse rostering problem.

To assist practitioner in designing successful MAs, a brief review of representative stochastic variation operators and individual learning schemes as the candidates for memetic operators is presented in the next subsections, followed by the discussion on several state-of-the-art coordination mechanisms of memetic operators that represents the recent advancing developments of memetic algorithms.

Bibliography

References

[1] U. Aickelin, E. K. Burke, and J. Li. An estimation of distribution algorithm with intelligent local search for rule-based nurse rostering. *Journal of the Operational Research Society*, 58(12):1574–1585, 2006. [This is a memetic approach EDA + Local Search for scheduling problems.]

[2] A. Auger and N. Hansen. A Restart CMA evolution strategy with increasing population size. Proceedings of the 2005 *IEEE Congress on Evolutionary Computation (CEC' 2005)*, 2:1769–1776, 2005. [This is a variant of CMAES. The algorithm performs a population restart and an increase of population size]

[3] A. Auger and N. Hansen. Performance Evaluation of an Advanced Local Search Evolutionary Algorithm. *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC' 2005)*, 2:1777–1784, 2005. [The performance of a restart CMAES is tested over CEC 2005]

[4] T. Bäck, D. B. Fogel, and Z. Michalewicz. *Handbook of evolutionary computation*. Taylor & Francis, 1997. [An edited book summarizing the advances in evolutionary algorithms in '80s and early '90s]

[5] J. E. Baker. Reducing bias and inefficiency in the selection algorithm. *Proceedings of the Second International Conference on Genetic Algorithms*, pp. 14–21, 1987. [An early contribution that discusses the selection mechanisms in genetic algorithms]

[6] S. Baluja. Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning. Technical report, Pittsburgh, PA, USA, 1994. [This report proposes a genetic algorithm that integrates a probabilistic criterion to generate new solutions]

[7] H. G. Beyer. *The Theory of Evolution Strategies*. Springer, 2001. [A monograph that analyses in depth theoretical aspects of evolution strategies]

[8] H.-G. Beyer and H.-P. Schwefel. Evolution strategies–A comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002. [This is a gentle but thorough survey on evolution strategies]

[9] A. Brindle. *Genetic algorithms for function optimization*. (doctoral dissertation and technical report tr81-2), Edmonton: University of Alberta, Department of Computer Science, 1981. [An early work on genetic algorithms and their general role]

[10] B. D. Bunday. *Basic Optimization Methods*. Edward Arnold (Publishers) Ltd, 1985. [A book reporting diverse optimization algorithms]

[11] E. K. Burke, G. Kendall, and E. Soubeiga. A tabu search hyperheuristic for timetabling and rostering. *Journal of Heuristics*, 9(6):451–470, 2003. [A hyperheuristic for scheduling problems]

[12] Y. Cai, X. Sun, and P. Jia. Probabilistic modeling for continuous eda with boltzmann selection and kullback-leibeler divergence. *Proceedings of the 8th annual conference on Genetic and Evolutionary Computation (GECCO '06)*, pp. 389–396, 2006. [An advanced and promising estimation of distribution algorithm]

[13] A. Caponio, G. L. Cascella, F. Neri, N. Salvatore, and M. Sumner. A fast adaptive memetic algorithm for on-line and off-line control design of pmsm drives. *IEEE Transactions on System Man and Cybernetics-part B, special issue on Memetic Algorithms*, 37(1):28–41, 2007. [This is a memetic algorithm with fitness diversity adaptation for a control engineering problem]

[14] A. Caponio, F. Neri, and V. Tirronen. Super-fit control adaptation in memetic differential evolution frameworks. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 13(8):811–831, 2009. [A memetic algorithm based on a DE framework with a diversity adaptation. The DE is discovered to work well when one individual is much above the average.]

[15] I. Carreras, D. Miorandi, R. Saint-Paul, and Chlamtac I. Bottom-Up Design Patterns and The Energy Web. *IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans*, 40(4):815–824, 2010. [This paper proposed the use of bottom-up pattern to build on results in complex systems science and systems biology, for devising new solutions for designing and managing such environments.]

[16] IEEE *Congress on Evolutionary Computation 2007*, Singapore, 2007. IEEE Press. [Important conference proceedings]

[17] K. Chakhlevitch and P. Cowling. Hyperheuristics: Recent developments. In C. Cotta, M. Sevaux, and K. S¨orensen, editors, *Adaptive and Multilevel Metaheuristics*, volume 136 of Studies in Computational Intelligence, pp. 3–29. Springer-Verlag, Berlin Heidelberg, 2008. [A thorough survey on hyperheuristics]

[18] U. K. Chakraborty. *Advances in Differential Evolution*. Springer Verlag, 2008. [An edited book with several DE schemes and variants]

[19] R. Chelouah and P. Siarry. Tabu search applied to global optimization. *European Journal of Operational Research*, 123(2):256–270, 2000. [This showes how Tabu search is not only a local search but can detect high quality solutions]

[20] R. Chelouah and P. Siarry. A hybrid method combining continuous tabu search and Nelder-Mead simplex algorithms for the global optimization of multiminima functions. *European Journal of Operational Research*, 161(3):636–654, 2005. [A memetic approach to solve multimodal problems]

[21] T. Chen, K. Tang, G. Chen, and X. Yao. On the analysis of average time complexity of estimation of distribution algorithms. *IEEE Congress on Evolutionary Computation, 2007. CEC 2007*, pp. 453–460, 2007. [This paper discusses and compares the time complexity in estimation of distribution algorithms]

[22] X. Chen, Y. S. Ong, and M. H. Lim. Cooperating memes for vehicle routing problems. *International Journal of Innovative Computing, Information and Control*, 7(11):1–10, 2011. [A memetic computing approach using a list of memes for vehicle routing problems]

[23] X. S. Chen. An algorithm development environment for problem-solving. In *International Conference on Computational Problem-Solving*, pp. 4031–4038, 2010. [A visionary theoretical work on algorithmic design]

[24] X. S. Chen and Y. S. Ong. A Conceptual Modeling of Meme Complexes in Stochastic Search. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications*, In Press, 2012. [A theoretical work that poses the basis for automatic design of optimization algorithms]

[25] X. S. Chen, Y. S. Ong, M. H. Lim, and K. C. Tan. A Multi-Facet Survey on Memetic Computation. *IEEE Transactions on Evolutionary Computation*, 15(5):591–607, 2011. [A thorough survey on memetic computing]

[26] D. C. Chin. Comparative study of stochastic algorithms for system optimization based on gradient approximations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 27(2):244–249, 1997. [This a survey paper aiming to compare stochastic optimization algorithms that perform gradient approximations]

[27] A. Corana, M. Marchesi, C. Martini, and S. Ridella. Minimizing multimodal functions of continuous variables with the simulated annealing algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 13(3):262–280, 1987. [This is a modified simulated annealing for continuous optimization]

[28] P. Cowling, G. Kendall, and E. Soubeiga. A hyperheuristic approach to scheduling a sales summit. In *Proceedings of the Third International Conference on Practice and Theory of Automated Timetabling, volume 2079 of Lecture Notes in Computer Science*, pp. 176–190. Springer, 2000. [An implementation of a hyperheuristic for a scheduling problem]

[29] W. Daneshyari and G. G. Yen. Cultural MOPSO: A cultural framework to adapt parameters of multiobjective particle swarm optimization. *2008 IEEE Congress on Evolutionary Computation, CEC 2008*, pp. 1325–1332, 2008. [A memetic/cultural approach based on particle swarm optimization for multi-objective problems]

[30] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995. MHS '95., pp. 39–43, 1995. [Initial paper that defines particle swarm optimization]

[31] R. Eberhart, P. Simpson, and R. Dobbins. *Computational intelligence PC tools*. Academic Press Professional, Inc., San Diego, CA, USA, 1996. [A comprehensive book on early results of computational intelligence]

[32] C. A. Floudas. *Deterministic Global Optimization: Theory, Methods, and Applications*. Springer-Verlag New York, 2005. [A book that collects deterministic optimization algorithms]

[33] C. C. García-Martínez and M. Lozano. Hybrid Real-Coded Genetic Algorithms with Female and Male Differentiation. *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC' 2005)*, pp. 896–903, 2005. [A memetic algorithm that divides the population into two categories]

[34] C. K. Goh, Y. S. Ong, and K. C. Tan. *Multi-Objective Memetic Algorithms*. Springer,Berlin, Heidelberg, 2009. [An edited book collecting recent approaches on multi-objective memetic algorithms]

[35] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Co., Reading, MA, USA, 1989. [A cornerstone book on genetic algorithms and machine learning]

[36] D. E. Goldberg and K. Deb. A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. *Foundations of Genetic Algorithms*, 1:69–93, 1991. [This compares various selection schemes of genetic algoritms]

[37] M. Gong, L. Jiao, and L. Zhang. Baldwinian learning in clonal selection algorithm for optimization. *Information Sciences*, 180(8):1218–1236, 2010. [A memetic approach using Baldwinian learning]

[38] N. Hansen and S. Kern. Evaluating the CMA evolution strategy on multimodal test functions. In *Parallel Problem Solving from Nature-PPSN VIII*, pp. 282–291. Springer, 2004. [An experimental work on CMAES]

[39] N. Hansen, S. D. Muller, and P. Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation*, 11(1):1–18, 2003. [A variant of CMAES]

[40] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001. [One of the original implementations of CMAES]

[41] W. E. Hart. *Adaptive Global Optimization with Local Search*. PhD thesis, University of California, San Diego, 1994. [A thesis on memetic algorithms as evolutionary frameworks incorporating local search]

[42] W. E. Hart, N. Krasnogor, and J. E. Smith. *Recent Advances in Memetic Algorithms*, volume 166. Springer, 2004. [An edited book collecting many implementations and applications of memetic algorithms]

[43] F. Herrera, M. Lozano, and J. L. Verdegay. Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioural Analysis. *Artificial Intelligence Review*, 12(4):265–319, 1998. [A comparative analysis of different encoding in genetic algorithms]

[44] M. W. Ho. Why Lamarck won't go away. *Annals of Human Genetics*, 60(1):81–84, 1996. [A biology paper discussing the Lamarckianism]

[45] J. H. Holland. *Adaptation in natural and artificial systems*. MIT Press Cambridge, MA, USA, 1992. [A cornerstone book on genetic algorithms]

[46] H. H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications*. Morgan Kaufmann, 2005. [a book on stochastic local search]

[47] C. Houck, J. Joines, and M. Kay. Utilizing Lamarckian evolution and the Baldwin effect in hybrid genetic algorithms. Technical report, 1996. [An early comparative study on memetic algorithms and the insertion of improved solutions]

[48] X. Hu, Y. Shi., and R. Eberhart. Recent advances in particle swarm. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC2004)*, pp. 90–97, 2004. [A short survey paper on particle swarm optimization]

[49] G. Iacca, F. Neri, E. Mininno, Y. S. Ong, and M. H. Lim. Ockham's razor in memetic computing: Three stage optimal memetic exploration. *Information Sciences*, 188:17–43, 2012. [A paper defining the lex pasimoniae in memetic computing and proving how simple approaches can be very competitive]

[50] H. Ishibuchi, T. Yoshida, and T. Murata. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE Transactions on*

Evolutionary Computation, 7(2):204–223, 2003. [An important contribution discussing the balance between global and local search with reference to multi-objective optimization]

[51] E. Jablonka and M. J. Lamb. *Epigenetic Inheritance and Evolution: The Lamarckian Dimension*. Oxford University Press, 1995. [A biology paper discussing Lamarckianism]

[52] Y. Jin. A Comprehensive Survey of Fitness Approximation in Evolutionary Computation. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 9(1):3–12, 2005. [A survey on approximated/surrogate models for computationally expensive problems]

[53] A. J. Keane. The OPTIONS Design Exploration System, 1995. [An explanation of a software package for design of optimization algorithms and for optimization problems]

[54] C. T. Kelley. *Solving Nonlinear Equations with Newton's Method*. in 'Fundamentals of Algorithms' series, SIAM Press., 2003. [A short applied mathematics book about Newton's method]

[55] G. Kendall, P. Cowling, and E. Soubeiga. Choice function and random hyperheuristics. In *Proceedings of the Fourth Asia-Pacific Conference on Simulated Evolution and Learning*, pp. 667–71, 2002. [A paper about coordination of search algorithms in hyper-heuristics]

[56] A. V. Kononova, K. J. Hughes, M. Pourkashanian, and D. B. Ingham. Fitness diversity based adaptive memetic algorithm for solving inverse problems of chemical kinetics. In CEC [16], pp. 2366–2373. [A memetic algorithm with fitness diversity coordination of the local search to address a problem in chemical kinetics]

[57] A. V. Kononova, D. B. Ingham, and M. Pourkashanian. Simple scheduled memetic algorithm for inverse problems in higher dimensions: Application to chemical kinetics. In *CEC 2008*, pp. 3906–3913, Hong Kong, 2008. IEEE Press.[A memetic algorithm hyper-heuristic coordination of the local search to address a problem in chemical kinetics]

[58] P. Korosec, J. Silc, and B. Filipic. The differential ant-stigmergy algorithm. *Information Sciences*, Volume 192, pp. 82–97, 2012 [A memetic approach combining ACO and DE]

[59] N. Krasnogor. Coevolution of genes and memes in memetic algorithms. In A. Wu, editor, *Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop Program*, 1999. [A coevolutionary/self-adaptive memetic approach]

[60] N. Krasnogor. *Studies on the Theory and Design Space of Memetic Algorithms*. PhD thesis, Doctoral dissertation, University of the West of England, Bristol, England, 2002. [This is a doctoral thesis containing important theoretical results on memetic algorithms]

[61] N. Krasnogor. Self-generating metaheuristics in bioinformatics: The protein structure comparison case. *Genetic Programming and Evolvable Machines*, 5(2):181–201, 2004. [A self-adaptive approach that evolves to generate optimization algorithms with reference to a biological problem]

[62] N. Krasnogor, B. P. Blackburne, E. K. Burke, and J. D. Hirst. Multimeme algorithms for protein structure prediction. *Proceedings of the Parallel Problem Solving from Nature VII. Lecture Notes in Computer Science*., 2439/2002:769–778, 2002. [A memetic scheme with multiple local search for protein structure prediction]

[63] N. Krasnogor and S. Gustafson. A study on the use of "self-generation" in memetic algorithms. *Natural Computing*, 3(1):53–76, 2004. [This is a study of self-adaptation as the evolutionary self-design of memetic algorithms]

[64] N. Krasnogor and J. Smith. A memetic algorithm with self-adaptive local search: TSP as a case study. In L. D. Whitley, D. Goldberg, E. Cantu-Paz, L. Spector, I. Parmee, H. G Beyer, editors, *GECCO 2000*, pp. 987–994, Las Vegas NV, 2000. Morgan Kaufmann. [This is a memetic approach for the TSP]

[65] N. Krasnogor and J. Smith. Emergence of profitable search strategies based on a simple inheritance mechanism. In L. Spector, E. Goodman, A. Wu, W. Langdon, H. M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M. Garzon, E. Burke, editors, *GECCO 2001*, pp. 432–439, San Francisco, California, USA, 2001. Morgan Kaufmann. [A theoretical work that shows how a simple inheritance mechanism can lead to a high performance]

[66] N. Krasnogor and J. Smith. A tutorial for competent memetic algorithms: Model, taxonomy, and design issues. *IEEE Transactions on Evolutionary Computation*, 9(5):474–488, 2005. [A survey that

defines a platform for memetic design]

[67] K. W. C. Ku, M. W. Mak, and W. C. Siu. A study of the Lamarckian evolution of recurrent neural networks. *IEEE Transactions on Evolutionary Computation*, 4(1):31–42, 2000. [A study on memetic algorithms with reference to neural network training]

[68] M. W. S. Land. *Evolutionary Algorithms with Local Search for Combinatorial Optimization*. PhD thesis, Citeseer, 1998. [A thesis investigation memetic algorithms in combinatorial optimization]

[69] P. Larranaga. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Kluwer Academic Publishers, 2001. [A book on estimation of distribution algorithms]

[70] M. N. Le. *Self-configurable Memetic Algorithm*. PhD thesis, Center for Computational Intelligence, School of Computer Engineering, Nanyang Technological University, Singapore, 2012. [A thesis investigating the possibility of automatic design in memetic algorithms]

[71] M. N. Le, Y. S. Ong, Y. Jin, and B. Sendhoff. Lamarckian memetic algorithms: local optimum and connectivity structure analysis. *Memetic Computing Journal*, 1(3):175–190, 2009. [A theoretical work on functioning of memetic algorithms and generation of candidate solutions]

[72] M. N. Le, Y. S. Ong, Y. Jin, and B. Sendhoff. A Unified Framework for Symbiosis of Evolutionary Mechanisms with Application to Water Clusters Potential Model Design. *IEEE Computational Intelligence Magazine*, 7(1):20–35, 2012. [This article presents a theoretical model for promoting the algorithmic search by means of a stochastic variation of the solution]

[73] C.-Y. Lee and X. Yao. Evolutionary programming using mutations based on the Lévy probability distribution. *IEEE Transactions on Evolutionary Computation*, 8(1):1–13, 2004. [A modified evolutionary programming using Lévy probability]

[74] W. F. Leong and G. G. Yen. Dynamic population size in PSO-based multiobjective optimization. *2006 IEEE Congress on Evolutionary Computation, CEC 2006*, pp. 1718–1725, 2006. [A PSO implementation for multi-objective optimization problems]

[75] W. F. Leong and G. G. Yen. PSO-based multiobjective optimization with dynamic population size and adaptive local archives. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(5):1270–1293, 2008. [An extended version of the algorithm in [74]]

[76] R. Lewis and B. Paechter. Finding feasible timetables using group-based operators. *IEEE Transactions on Evolutionary Computation*, 11(3):397–413, 2007. [This paper investigates the applicability of group-based genetic algorithms to the university course timetabling problem.]

[77] R. M. Lewis, V. Torczon, and M. W. Trosset. Direct search methods: then and now. *Journal of Computational and Applied Mathematics*, 124(1-2):191–207, 2000. [A survey on direct search methods spanning from methods proposed in '60s until early '70s]

[78] B. Li, Y. S. Ong, M. N. Le, and C. K. Goh. Memetic Gradient Search. *IEEE Congress on Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence)*, 1-6:2894 – 2901, June 2008. [A review on different gradient-based methods]

[79] D. Lim, Y. Jin, Y. S. Ong, and B. Sendhoff. Generalizing Surrogate-assisted Evolutionary Computation. *IEEE Transactions on Evolutionary Computation*, 14(3):329–355, June 2010. [This paper proposes a framework for optimization in the case of computationally expensive problems. This framework uses multiple approximate models (surrogate) to evolve towards a flexible solver]

[80] M. Lozano, F. Herrera, N. Krasnogor, and D. Molina. Real-coded Memetic Algorithms with Crossover Hill-climbing. *Evolutionary Computation*, 12(3):273–302, 2004. [This paper proposes a memetic algorithm for continuous problems based on a genetic algorithm and a crossover within the solution generation]

[81] R. Mallipeddi, S. Mallipeddi, and P. N. Suganthan. Ensemble strategies with adaptive evolutionary programming. *Information Sciences*, 180(9):1571–1581, 2010. [This paper proposes an evolutionary programming framework enriched with multiple mutations that are selected on the basis of their success]

[82] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing*, 11(2):1679–1696, 2011. [This is a DE framework with a pool of mutation and a variation mechanism for the parameters. The

selection is performed by a mechanism that rewards the most successful combinations]

[83] J. J. Merelo Guervós, P. Adamidis, H.G. Beyer editors. *Parallel Problem Solving from Nature VII, volume 2439 of Lecture Notes in Computer Science*, Granada, Spain, 2002. Springer-Verlag. [A conference proceedings]

[84] P. *Merz. Memetic algorithms for combinatorial optimization problems: fitness landscapes and effective search strategies*. PhD thesis, University of Siegen, Germany, 2000. [This thesis studies combinatorial problems, analyses search components in the light of memetic hybridisation]

[85] P. Merz. Advanced fitness landscape analysis and the performance of memetic algorithms. *Evolutionary Computation*, 12(3):303–325, 2004. [A theoretical paper that analyses the fitness landscape features, with reference to combinatorial optimization, to select the proper memetic operators]

[86] R. Meuth, M. H. Lim, Y. S. Ong, and D. C. Wunsch II. A Proposition on Memes and Meta-memes in computing for higher-order learning. *Memetic Computing*, 1(2):85–100, 2009. [An abstract analysis on self-generation of optimization algorithms]

[87] Z. Michalewicz. *Genetic Algorithms+ Data Structures= Evolution Programs*. Springer-Verlag London, UK, 1996. [A cornerstone book on evolutionary computation]

[88] H. Miihlenbein and G. Paafi. From recombination of genes to the estimation of distributions I. Binary parameters. *Parallel Problem Solving from Nature-PPSN IV*, pp. 178–187, 1996. [A comparative study on recombination strategies]

[89] D. Molina, F. Herrera, and M. Lozano. Adaptive Local Search Parameters for Real-Coded Memetic Algorithms. *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC' 2005)*, pp. 888–895, 2005. [This paper proposes an adaptive memetic algorithm that uses an adaptive scheme to decide the occurrence and depth of local search]

[90] D. Molina, M. Lozano, C. García-Martínez, and F. Herrera. Memetic Algorithm for Intense Local Search Methods Using Local Search Chains. *Hybrid Metaheuristics*, 5296:58–71, 2008. [This paper proposes an adaptive scheme in memetic framework that, repeatedly and in different stages, uses a local search on the same solutions]

[91] D. Molina, M. Lozano, C. García-Martínez, and F. Herrera. Memetic Algorithms for Continuous Optimisation Based on Local Search Chains. *Evolutionary Computation*, 18(1):27–63, 2010. [This paper proposes a memetic framework composed on a genetic algorithm and multiple CMAES instances activated and continued by a heuristic rule]

[92] P. Moscato. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. *Caltech Concurrent Computation Program*, C3P Report, 826:1989, 1989. [A report that introduced at first the term memetic algorithm]

[93] P. Moscato. Memetic algorithms: a short introduction. *Mcgraw-Hill's Advanced Topics In Computer Science Series*, pp. 219–234, 1999. [An overview on memetic algorithms, especially on the origins and definitions]

[94] C. L. Muller, B. Baumgartner, and I. F. Sbalzarini. Particle Swarm CMA Evolution Strategy for the Optimization of Multi-Funnel Landscapes. *Proceedings of the 2009 IEEE Congress on Evolutionary Computation (CEC' 2009)*, pp. 2685–2692, 2009. [A paper that proposes a memetic algorithm composed of a PSO framework and multiple CMAES instances that work on different candidate solutions]

[95] A. Nelder and R. Mead. A simplex method for function optimization. *Computation Journal*, Vol 7:308–313, 1965. [This is a famous gradient-free optimization algorithm]

[96] F. Neri, C. Cotta, and P. Moscato. *Handbook of Memetic Algorithms, volume 379 of Studies in Computational Intelligence*. Springer, 2012. [An edited book reviewing recent works in memetic computing]

[97] F. Neri, G. Iacca, and E. Mininno. Disturbed exploitation compact differential evolution for limited memory optimization problems. *Information Sciences*, 181(12):2469–2487, 2011. [A compact memetic approach based on a compact differential evolution and a randomized re-start mecanism]

[98] F. Neri, V. Tirronen, T. Kärkkäinen, and T. Rossi. Fitness diversity based adaptation in multimeme algorithms: A comparative study. In CEC [16], pp. 2374–2381. [A comparative study on fitness diversity

adaptation schemes]

[99] F. Neri, J. Toivanen, G. L. Cascella, and Y. S. Ong. An adaptive multimeme algorithm for designing HIV multidrug therapies. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(2):264–278, 2007. [A memetic algorithm with multiple local search components and a fitness diversity adaptation for a large scale and computationally expensive combinatorial problem with application to HIV]

[100] F. Neri, J. Toivanen, and R. A. E. Mäkinen. An adaptive evolutionary algorithm with intelligent mutation local searchers for designing multidrug therapies for HIV. *Applied Intelligence*, 27(3):219–235, December 2007. [A memetic algorithm with multiple intelligent operators to solve the optimal HIV therapy problem with binary representation]

[101] Q. H. Nguyen, Y. S. Ong, and N. Krasnogor. A study on the design issues of Memetic Algorithm. *IEEE Congress on Evolutionary Computation (CEC), 25-28 September 2007*, pp. 2390–2397, 2007. [This paper discusses relevant design issues in memetic algorithms and shows the impact on the performance for some algorithmic choices]

[102] Q. H. Nguyen, Y. S. Ong, and M. H. Lim. Non-genetic transmission of memes by diffusion. Pp. 1017–1024, 2008. [A discussion about the role of memes and the proposal of the diffusion memetic algorithm where the memetic material is transmitted by means of non-genetic transfer.]

[103] Q. H. Nguyen, Y. S. Ong, and M. H. Lim. A Probabilistic Memetic Framework. *IEEE Transactions on Evolutionary Computation*, 13:604–623, 2009. [This paper proposes a memetic framework that models the algorithm as a decision process. The framework has to decide the action to take (within evolution and individual learning) on the basis of a probabilistic analysis of each process in locating the global optimum]

[104] Q. H. Nguyen, Y. S. Ong, M. H. Lim, and N. Krasnogor. Adaptive Cellular Memetic Algorithms. *Evolutionary Computation*, 17(2):231–256, 2009. [In this paper, a memetic algorithms with structured population is proposed.]

[105] N. Noman and H. Iba. Accelerating Differential Evolution using an Adaptive Local Search. *IEEE Transactions on Evolutionary Computation,* 12(1):107–125, 2008. [A DE implementation with a simplex local search in the crossover]

[106] Y. S. Ong and A. J. Keane. Meta-Lamarckian Learning in Memetic Algorithms. *IEEE Transactions on Evolutionary Computation*, 8(2):99–110, 2004. [This paper proposes a novel success-based adaptation mechanism for the selection of local search components in memetic frameworks]

[107] Y. S. Ong, M. H. Lim, and X. S. Chen. Memetic Computation - Past, Present & Future [Research Frontier]. *Computational Intelligence Magazine, IEEE*, 5(2):24–31, 2010. [A survey on memetic computing with suggestions about the future trends of the subject toward self-generation of the algorithms]

[108] Y. S. Ong, M. H. Lim, N. Zhu, and K. W. Wong. Classification of adaptive memetic algorithms: A comparative study. *IEEE Transactions On Systems, Man and Cybernetics - Part B*, 36(1):141–152, 2006. [A survey and taxonomy on adaptation mechanisms in memetic algorithms]

[109] Y. S. Ong, P. B. Nair, and A. J. Keane. Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modelling. *American Institute of Aeronautics and Astronautics Journal*, 41(4):687–696, 2003. [A study on surrogate-assisted models]

[110] Y. S. Ong, P. B. Nair, and K. Lum. Max-min surrogate-assisted evolutionary algorithm for robust design. *IEEE Transactions on Evolutionary Computation*, 10(4):392–404, 2006. [A study on robust genetic algorithms to large scale optimization problems]

[111] I. Paenke, Y. Jin, and J. Branke. Balancing population and individual level adaptation in changing environments. *Adaptive Behavior*, 17(2):153–174, 2009. [This paper discusses the balance of the binomial evolution vs local search and investigates about a suitable trade-off]

[112] I. Paenke, B. Sendhoff, J. Rowe, and C. Fernando. On the adaptive disadvantage of Lamarckianism in rapidly changing environments. *Advances in Artificial Life*, pp. 355–364, 2007. [This paper discusses about the performance of memetic alforithms in dynamic optimization problems. The Lamarckian schemes seem not to be suitable in this case.]

[113] K. E. Parsopoulos and M. N. Vrahatis. Recent approaches to global optimization problems through Particle Swarm Optimization. *Natural Computing*, 1(2):235–306, 2002. [A review paper on PSO]

[114] M. Pelikan, D. E. Goldberg, and E. Cantu-Paz. BOA: The Bayesian optimization algorithm. 1:525–532, 1999. [This paper proposes a famous estimation of distribution algorithm]

[115] M. Pelikan, D. E. Goldberg, and F. G. Lobo. A Survey of Optimization by Building and Using Probabilistic Models. *Computational optimization and applications*, 21(1):5–20, 2002. [A review paper with theoretical and algorithmic considerations on estimation of distribution algorithms]

[116] F. Peng, K. Tang, G. Chen, and X. Yao. Population-based algorithm portfolios for numerical optimization. *IEEE Transactions on Evolutionary Computation*, 14(5):782–800, 2010. [This paper proposes a portfolio algorithm, i.e. multiple algorithms that are scheduled to concurrently solve problems]

[117] R. Poli, J. Kennedy, and T. Blackwell. Particle Swarm Optimization: An overview. *Swarm Intelligence*, 1(1):33–57, 2007. [A review paper on PSO]

[118] M. J. D. Powell. An Efficient Method for Finding the Minimum of a Function of Several Variables without calculating derivatives. *Computer Journal*, 7(2):155–162, 1964. [A famous deterministic local search algorithm]

[119] K. V. Price, R. M. Storn, and J. A. Lampinen. *Differential evolution: a practical approach to global optimization*. Springer Verlag, 2005. [An edited book on foundations and applications of DE]

[120] J. M. Renders and H. Bersini. Hybridizing genetic algorithms with hill-climbing methods for global optimization: two possible ways. *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence.*, 1:312–317, 1994. [An early memetic approach presented as a hybrid genetic algorithm]

[121] J. Ronkkonen, S. Kukkonen, and K. Price. Real-Parameter Optimization with Differential Evolution. *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC' 2005)*, pp. 506–513, 2005. [DE is used to take part in a competition on numerical optimization]

[122] H. H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960. [A famous deterministic local search algorithm]

[123] T. Sasaki and M. Tokoro. Adaptation toward changing environments: Why Darwinian in nature. *European conference on Artificial Life*, pp. 145–153, 1997. [This paper is about multi-agent systems and studies learning and evolution within a population of neural networks. Darwinian and Lamarckian learning systems are investigated in dependence of static and dynamic environments]

[124] H.-P. Schwefel. Evolution and Optimum Seeking: *The Sixth Generation*. John Wiley & Sons, Inc., New York, NY, USA, 1993. [A cornerstone book on evolution strategies]

[125] B. Sendhoff and M. Kreutz. A model for the dynamic interaction between evolution and learning. *Neural Process. Lett.*, 10:181–193, December 1999. [This paper investigates the interaction between evolution and learning in memetic computing and proposes a model to describe such interactions]

[126] P. Siarry and G. Berthiau. Fitting of tabu search to optimize functions of continuous variables. *International Journal for Numerical Methods in Engineering*, 40(13):2449–2457, 1997. [This paper proposes an adaptation for tabu search]

[127] P. Siarry, G. Berthiau, F. Durdin, and J. Haussy. Enhanced simulated annealing for globally minimizing functions of many-continuous variables. *ACM Transactions on Mathematical Software (TOMS)*, 23(2):209–228, 1997. [A simulated annealing version to tackle large scale optimization problems]

[128] J. Smith. Co-evolution of memetic algorithms: Initial investigations. In Merelo Guervós, Adamidis, Beyer [83], pp. 537–548. [An initial study on co-evolving memetic algorithms, i.e. a parallel evolution of memes and solutions.]

[129] J. Smith. Protein structure prediction with co-evolving memetic algorithms. In *CEC 2003*, pp. 2346–2353. IEEE Press, Canberra, Australia, 2003. [An application of co-evolving memetic algorithms to protein structure prediction]

[130] J. Smith. The co-evolution of memetic algorithms for protein structure prediction. In *Recent*

advances in memetic algorithms, volume 166 of Studies in Fuzziness and Soft Computing, pp. 105–128. Springer-Verlag, 2005. [A review chapter on co-evolving memetic algorithms applied to protein structure predction]

[131] J. Smith, Y. S. Ong, S. Gustafson, M. H. Lim, and N. Krasnogor. Workshop on Memetic Algorithms. *IEEE Symposium Series on Computational Intelligence*, 2009. [Conference proceedings on memetic algorithms]

[132] J. E. Smith. Coevolving Memetic Algorithms: A Review and Progress Report. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(1):6–17, 2007. [A review paper on co-evolving memetic algorithms]

[133] F. J. Solis and R. J.-B. Wets. Minimization by Random Search Techniques. *Mathematics of Operations Research*, 6(1):pp. 19–30, 1981. [A famous deterministic local search algorithm]

[134] J. C. Spall. A stochastic approximation algorithm for large-dimensional systems in the Kiefer-Wolfowitz setting. *Proceedings of the 27th IEEE Conference on Decision and Control*, 2:1544–1548, 1988. [A famous stochastic gradient-free local search]

[135] J. C. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341, 1992. [A famous stochastic gradient-free algorithm that performs a gradient approximation on the basis of sampled points]

[136] J. C. Spall. *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, April 2003. [A book on stochastic local search]

[137] R. Storn and K. Price. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4):341–359, 1997. [One of the earliest implementation of DE]

[138] J. Tang, M. H. Lim, and Y. S. Ong. Diversity-Adaptive Parallel Memetic Algorithm for Solving Large Scale Combinatorial Optimization Problems. *Soft Computing Journal*, 11(9):873–888, 2007. [This paper proposes a memetic algorithm with structured population for large scale optimization problems]

[139] K. Tang, Y. Mei, and X. Yao. Memetic Algorithm with Extended Neighborhood Search for Capacitated Arc Routing Problems. *IEEE Transactions on Evolutionary Computation*, 13(5):1151–1166, 2009. [The NP-hard problem of capacitated arc routing problem is tackled by means of a memetic approach]

[140] V. Tirronen, F. Neri, T. Kärkkäinen, K. Majava, and T. Rossi. An enhanced memetic differential evolution in filter design for defect detection in paper production. *Evolutionary Computation*, 16:529–555, 2008. [A memetic algorithm based on differential evolution and multiple local search coordinated by a fitness diversity adaptation]

[141] M. Ulieru and D. R. Emergent Engineering: A Radical Paradigm Shift. *Int. J. Autonomous and Adaptive Communications Systems*, 4(1):39–60, 2011. [This paper proposes a techno-social system and infrastructures, the so-called cyber-physical ecosystems]

[142] D. Vanderbilt and S. G. Louie. A Monte Carlo simulated annealing approach to optimization over continuous variables. *Journal of Computational Physics*, 56(2):259–271, 1984. [An early work on simulated annealing]

[143] Z. Wang, K. Tang, and X. Yao. A Memetic Algorithm for Multi-Level Redundancy Allocation. *IEEE Transactions on Reliability*, 59(4):754–765, 2010. [A memetic algorithm for an NP-hard problem]

[144] D. Whitley, V. S. Gordon, and K. Mathias. Lamarckian evolution, the Baldwin effect and function optimization. *Parallel Problem Solving from Nature–PPSN III: International Conference on Evolutionary Computation, The Third Conference on Parallel Problem Solving from Nature*, pp. 6–15, 1994. [This paper discusses and compares two hybridization techniques in memetic algorithms: Lamarckian and Baldwinian evolution]

[145] D. Wolpert and W. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997. [Two theorems that state that the average performance of all the algorithms over all the possible problems is the same]

[146] A. H. Wright. Genetic algorithms for real parameter optimization. *Foundations of Genetic*

Algorithms, 1:205–218, 1991. [A review chapter on the early real-coded genetic algorithms]

[147] Z. Yang, K. Tang, and X. Yao. Differential Evolution for High-dimensional Function Optimization. *IEEE Congress on Evolutionary Computation, 2007. CEC 2007*, pp. 3523–3530, 2007. [Two DE based algorithms for with cooperative coevolution with reference to large scale problems]

[148] Z. Yang, K. Tang, and X. Yao. Scalability of generalized adaptive differential evolution for large-scale continuous optimization. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 2010. [A generalized parameter adaptation scheme for DE in large scale optimization]

[149] X. Yao. Simulated annealing with extended neighborhood. *International Journal of Computer Mathematics*, 40:169–189, 1991. [A change in the neighborhood generation in simulated annealing leads to an improvement in the algorithm speed]

[150] X. Yao and Y. Liu. Fast Evolutionary Programming. *Proceedings of the Fifth Annual Conference on Evolutionary Programming*, pp. 451–460, 1996. [A modified evolutionary programming]

[151] X. Yao, Y. Liu, and G. Lin. Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3:82–102, 1999. [An extended version of the modified evolutionary programming making use of the Cauchy distribution]

[152] G. G. Yen and W. F. Leong. Dynamic multiple swarms in multiobjective particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 39(4):890–911, 2009. [A dynamic population structuring in PSO context for multi-objective optimization]

[153] B. Yuan and M. Gallagher. Experimental Results for the Special Session on Real-Parameter Optimization at CEC 2005: A Simple, Continuous EDA. *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC' 2005)*, pp. 1792–1799, 2005. [An estimation of distribution algorithm to tackle continuous optimization problems]

[154] Q. Yuan, F. Qian, and W. Du. A hybrid genetic algorithm with the Baldwin effect. *Information Sciences*, 180(5):640–652, 2010. [This paper proposes an implementation of a hybrid algorithm where the local search coordination is performed by a Baldwinian effect. A theorem showing the performance is also given.]

Biographical Sketches

Yew-Soon Ong received the BS and MS degrees in electrical and electronics engineering from Nanyang Technological University (NTU), Singapore, in 1998 and 1999, respectively. He received the PhD degree on artificial intelligence in complex design from the Computational Engineering and Design Center, University of Southampton, UK in 2003. He is currently an Associate Professor and Director of the Center for Computational Intelligence at the School of Computer Engineering, NTU. Dr. Ong is the founding Technical Editor-in-Chief of Memetic Computing Journal, Chief Editor of the Springer book series on studies in adaptation, learning, and optimization, Associate Editor of the IEEE Computational Intelligence Magazine, the IEEE Transactions on Systems, Man and Cybernetics – Part B and several others. He is the Chair of the IEEE Computational Intelligence Society Intelligent System & Applications Technical Committee and has served as a guest editor of several journals, including the IEEE Transactions on Evolutionary Computation. His research work on Memetic computation was featured by Thomson Scientific's Essential Science Indicators as one of the most cited emerging area of research in August 2007. Recently, he also received the 2012 IEEE Transactions on Evolutionary Computation 'Outstanding Paper Award' for his work pertaining to the modeling of Probabilistic Memetic Framework. His current research interests in computational intelligence span across memetic computation, evolutionary optimization using approximation/surrogate/meta-models, complex design optimization, intelligent agents and machine learning.

Ferrante Neri obtained his first PhD in Electro-technical Engineering, from the Technical University of Bari, Italy, in Apr 2007. In Nov 2007, he obtained a second PhD in Computer Science from the University of Jyväskylä (UJ), Finland. At this time Ferrante was appointed as a Senior Research Assistant in Simulation and Optimization at UJ. In 2009 he was awarded an Academy Research Fellowship by the Academy of Finland in order to work on the project 'Algorithmic Design Issues in Memetic Computing'.

In 2010, UJ awarded him the title of Adjunct Professor in Computational Intelligence. In 2012 Ferrante was appointed as Reader in Computational Intelligence and in 2013 Professor of Computational Intelligence Optimisation at the Centre for Computational intelligence, De Montfort University, UK. Ferrante co-authored over 100 international scientific articles and one book. His current research interests include computational intelligence optimisation and more specifically memetic computing, differential evolution, noisy and large scale optimisation, and compact and parallel algorithms.

Minh Nghia Le received the B. Eng. degree in Computer Engineering and the PhD degree on self-configurable optimization for complex design from the School of Computer Engineering, Nanyang Technological University (NTU), Singapore, in 2006 and 2012, respectively. His current research interests include memetic computing, artificial intelligence, large scale optimization and data-driven research.