Create account Log in

Article Talk Read Edit More Search

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Deutsch
Français
한국어
ไทย
Edit links

Memetic algorithm

From Wikipedia, the free encyclopedia

This article **provides insufficient context for those unfamiliar with the subject**. Please help improve the article with a good introductory style. *(February 2011)*

Memetic algorithms (MA) represent one of the recent growing areas of research in evolutionary computation. The term MA is now widely used as a synergy of evolutionary or any population-based approach with separate individual learning or local improvement procedures for problem search. Quite often, MA are also referred to in the literature as Baldwinian evolutionary algorithms (EA), Lamarckian EAs, cultural algorithms, or genetic local search.

Contents [hide]

Introduction [edit]

Inspired by both Darwinian principles of natural evolution and Dawkins' notion of a meme, the term “Memetic Algorithm” (MA) was introduced by Moscato in his technical report[1] in 1989 where he viewed MA as being close to a form of population-based hybrid genetic algorithm (GA) coupled with an individual learning procedure capable of performing local refinements. The metaphorical parallels, on the one hand, to Darwinian evolution and, on the other hand, between memes and domain specific (local search) heuristics are captured within memetic algorithms thus rendering a methodology that balances well between generality and problem specificity. This two-stage nature makes them a special case of Dual-phase evolution.

In a more diverse context, memetic algorithms are now used under various names including Hybrid Evolutionary Algorithms, Baldwinian Evolutionary Algorithms, Lamarckian Evolutionary Algorithms, Cultural Algorithms, or Genetic Local Search. In the context of complex optimization, many different instantiations of memetic algorithms have been reported across a wide range of application domains, in general, converging to high-quality solutions more efficiently than their conventional evolutionary counterparts.

In general, using the ideas of memetics within a computational framework is called "Memetic Computing or Memetic Computation" (MC).[2][3] With MC, the traits of Universal Darwinism are more appropriately captured. Viewed in this perspective, MA is a more constrained notion of MC. More specifically, MA covers one area of MC, in particular dealing with areas of evolutionary algorithms that marry other deterministic refinement techniques for solving optimization problems. MC extends the notion of memes to cover conceptual entities of knowledge-enhanced procedures or representations.

The development of MAs [edit]

1st generation [edit]

The first generation of MA refers to hybrid algorithms, a marriage between a population-based global search (often in the form of an evolutionary algorithm) coupled with a cultural evolutionary stage. This first generation of MA although encompasses characteristics of cultural evolution (in the form of local refinement) in the search cycle, it may not qualify as a true evolving system according to Universal Darwinism, since all the core principles of inheritance/memetic transmission, variation, and selection are missing. This suggests why the term MA stirred up criticisms and controversies among researchers when first introduced.[1]

Pseudo code:

```
Procedure Memetic Algorithm
Initialize: Generate an initial population;
while Stopping conditions are not satisfied do
    Evaluate all individuals in the population.
    Evolve a new population using stochastic search operators.
    Select the subset of individuals, $\Omega_{il}$, that should undergo the individual improvement procedure.
    for each individual in $\Omega_{il}$ do
        Perform individual learning using meme(s) with frequency or probability of $f_{il}$, for a period of $t_{il}$.
        Proceed with Lamarckian or Baldwinian learning.
    end for
end while
```

2nd generation [edit]

Multi-meme,[4] Hyper-heuristic[5][6] and Meta-Lamarckian MA[7] are referred to as second generation MA exhibiting the principles of memetic transmission and selection in their design. In Multi-meme MA, the memetic material is encoded as part of the genotype. Subsequently, the decoded meme of each respective individual/chromosome is then used to perform a local refinement. The memetic material is then transmitted through a simple inheritance mechanism from parent to offspring(s). On the other hand, in hyper-heuristic and meta-Lamarckian MA, the pool of candidate memes considered will compete, based on their past merits in generating local improvements through a reward mechanism, deciding on which meme to be selected to proceed for future local refinements. Memes with a higher reward have a greater chance of being replicated or copied. For a review on second generation MA; i.e., MA considering multiple individual learning methods within an evolutionary system, the reader is referred to.[8]

3rd generation [edit]

Co-evolution[9] and self-generating MAs[10] may be regarded as 3rd generation MA where all three principles satisfying the definitions of a basic evolving system have been considered. In contrast to 2nd generation MA which assumes that the memes to be used are known a priori, 3rd generation MA utilizes a rule-based local search to supplement candidate solutions within the evolutionary system, thus capturing regularly repeated features or patterns in the problem space.

Some design notes [edit]

The frequency and intensity of individual learning directly define the degree of evolution (exploration) against individual learning (exploitation) in the MA search, for a given fixed limited computational budget. Clearly, a more intense individual learning provides greater chance of convergence to the local optima but limits the amount of evolution that may be expended without incurring excessive computational resources. Therefore, care should be taken when setting these two parameters to balance the computational budget available in achieving maximum search performance. When only a portion of the population individuals undergo learning, the issue on which subset of individuals to improve need to be considered to maximize the utility of MA search. Last but not least, the individual learning procedure/meme used also favors a different neighborhood structure, hence the need to decide which meme or memes to use for a given optimization problem at hand would be required.

How often should individual learning be applied? [edit]

One of the first issues pertinent to memetic algorithm design is to consider how often the individual learning should be applied; i.e., individual learning frequency. In one case,[11] the effect of individual learning frequency on MA search performance was considered where various configurations of the individual learning frequency at different stages of the MA search were investigated. Conversely, it was shown elsewhere[12] that it may be worthwhile to apply individual learning on every individual if the computational complexity of the individual learning is relatively low.

On which solutions should individual learning be used? [edit]

On the issue of selecting appropriate individuals among the EA population that should undergo individual learning, fitness-based and distribution-based strategies were studied for adapting the probability of applying individual learning on the population of chromosomes in continuous parametric search problems with Land[13] extending the work to combinatorial optimization problems. Bambha et al. introduced a simulated heating technique for systematically integrating parameterized individual learning into evolutionary algorithms to achieve maximum solution quality.[14]

How long should individual learning be run? [edit]

Individual learning intensity, t_{il}, is the amount of computational budget allocated to an iteration of individual learning; i.e., the maximum computational budget allowable for individual learning to expend on improving a single solution.

What individual learning method or meme should be used for a particular problem or individual? [edit]

In the context of continuous optimization, individual learning/individual learning exists in the form of local heuristics or conventional exact enumerative methods.[15] Examples of individual learning strategies include the hill climbing, Simplex method, Newton/Quasi-Newton method, interior point methods, conjugate gradient method, line search, and other local heuristics. Note that most of common individual learninger are deterministic.

In combinatorial optimization, on the other hand, individual learning methods commonly exist in the form of heuristics (which can be deterministic or stochastic) that are tailored to a specific problem of interest. Typical heuristic procedures and schemes include the k-gene exchange, edge exchange, first-improvement, and many others.

Applications [edit]

Memetic algorithms have been successfully applied to a multitude of real-world problems. Although many people employ techniques closely related to memetic algorithms, alternative names such as *hybrid genetic algorithms* are also employed. Furthermore, many people term their memetic techniques as *genetic algorithms*.[*citation needed*]

Researchers have used memetic algorithms to tackle many classical NP problems. To cite some of them: graph partitioning, multidimensional knapsack, travelling salesman problem, quadratic assignment problem, set cover problem, minimal graph coloring, max independent set problem, bin packing problem, and generalized assignment problem.

More recent applications include (but are not limited to) training of artificial neural networks,[16] pattern recognition,[17] robotic motion planning,[18] beam orientation,[19] circuit design,[20] electric service restoration,[21] medical expert systems,[22] single machine scheduling,[23] automatic timetabling (notably, the timetable for the NHL),[24] manpower scheduling,[25] nurse rostering optimisation,[26] processor allocation,[27] maintenance scheduling (for example, of an electric distribution network),[28] multidimensional knapsack problem,[29] VLSI design,[30] clustering of gene expression profiles,[31] feature/gene selection,[32][33] and multi-class, multi-objective feature selection.[34][35]

Recent Activities in Memetic Algorithms [edit]

- IEEE Workshop on Memetic Algorithms (WOMA 2009). Program Chairs: Jim Smith, University of the West of England, U.K.; Yew-Soon Ong, Nanyang Technological University, Singapore; Gustafson Steven, University of Nottingham; U.K.; Meng Hiot Lim, Nanyang Technological University, Singapore; Natalio Krasnogor, University of Nottingham, U.K.
- Memetic Computing Journal, first issue appeared in January 2009.
- 2008 IEEE World Congress on Computational Intelligence (WCCI 2008), Hong Kong, Special Session on Memetic Algorithms.
- Special Issue on 'Emerging Trends in Soft Computing - Memetic Algorithm', Soft Computing Journal, Completed & In Press, 2008.
- IEEE Computational Intelligence Society Emergent Technologies Task Force on Memetic Computing
- IEEE Congress on Evolutionary Computation (CEC 2007), Singapore, Special Session on Memetic Algorithms.
- 'Memetic Computing' by Thomson Scientific's Essential Science Indicators as an Emerging Front Research Area.

- Special Issue on Memetic Algorithms, IEEE Transactions on Systems, Man and Cybernetics - Part B, Vol. 37, No. 1, February 2007.
- Recent Advances in Memetic Algorithms, Series: Studies in Fuzziness and Soft Computing, Vol. 166, ISBN 978-3-540-22904-9, 2005.
- Special Issue on Memetic Algorithms, Evolutionary Computation Fall 2004, Vol. 12, No. 3: v-vi.

References [edit]

1. ^ a b Moscato, P. (1989). "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms". *Caltech Concurrent Computation Program* (report 826).
2. ^ Chen, X. S.; Ong, Y. S.; Lim, M. H.; Tan, K. C. (2011). "A Multi-Facet Survey on Memetic Computation". *IEEE Transactions on Evolutionary Computation* **15** (5): 591–607. doi:10.1109/tevc.2011.2132725.
3. ^ Chen, X. S.; Ong, Y. S.; Lim, M. H. (2010). "Research Frontier: Memetic Computation - Past, Present & Future". *IEEE Computational Intelligence Magazine* **5** (2): 24–36. doi:10.1109/mci.2010.936309.
4. ^ Krasnogor N. (1999). "Coevolution of genes and memes in memetic algorithms". *Graduate Student Workshop*: 371.
5. ^ Kendall G. and Soubeiga E. and Cowling P. *Choice function and random hyperheuristics*. 4th Asia-Pacific Conference on Simulated Evolution and Learning. SEAL 2002. pp. 667–671.
6. ^ Burke E. K. and Gendreau M. and Hyde M. and Kendall G. and Ochoa G. and Özcan E. and Qu R. (2013). "Hyper-heuristics: A Survey of the State of the Art". *Journal of the Operational Research Society* **64** (12): 1695–1724.
7. ^ Ong Y. S. and Keane A. J. (2004). "Meta-Lamarckian learning in memetic algorithms". *IEEE Transactions on Evolutionary Computation* **8** (2): 99–110. doi:10.1109/TEVC.2003.819944.
8. ^ Ong Y. S. and Lim M. H. and Zhu N. and Wong K. W. (2006). "Classification of Adaptive Memetic Algorithms: A Comparative Study". *IEEE Transactions on Systems Man and Cybernetics – Part B.* **36** (1): 141. doi:10.1109/TSMCB.2005.856143.
9. ^ Smith J. E. (2007). "Coevolving Memetic Algorithms: A Review and Progress Report". *IEEE Transactions on Systems Man and Cybernetics - Part B* **37** (1): 6–17. doi:10.1109/TSMCB.2006.883273.
10. ^ Krasnogor N. and Gustafson S. (2002). "Toward truly "memetic" memetic algorithms: discussion and proof of concepts". *Advances in Nature-Inspired Computation: the PPSN VII Workshops. PEDAL (Parallel Emergent and Distributed Architectures Lab). University of Reading*.
11. ^ Hart W. E. (1994). "Adaptive Global Optimization with Local Search". University of California.
12. ^ Ku K. W. C. and Mak M. W. and Siu W. C. (2000). "A study of the Lamarckian evolution of recurrent neural networks". *IEEE Transactions on Evolutionary Computation* **4** (1): 31–42. doi:10.1109/4235.843493.
13. ^ Land M. W. S. (1998). "Evolutionary Algorithms with Local Search for Combinatorial Optimization". UNIVERSITY OF CALIFORNIA.
14. ^ Bambha N. K. and Bhattacharyya S. S. and Teich J. and Zitzler E. (2004). "Systematic integration of parameterized local search into evolutionary algorithms". *IEEE Transactions on Evolutionary Computation* **8** (2): 137–155. doi:10.1109/TEVC.2004.823471.
15. ^ Schwefel H. P. (1995). *Evolution and optimum seeking*. Wiley New York.
16. ^ Ichimura, T.; Kuriyama, Y. (1998). *Learning of neural networks with parallel hybrid GA using a royal road function*. IEEE International Joint Conference on Neural Networks **2**. New York, NY. pp. 1131–1136.
17. ^ Aguilar, J.; Colmenares, A. (1998). "Resolution of pattern recognition problems using a hybrid genetic/random neural network learning algorithm". *Pattern Analysis and Applications* **1** (1): 52–61. doi:10.1007/BF01238026.
18. ^ Ridao, M.; Riquelme, J.; Camacho, E.; Toro, M. (1998). "An evolutionary and local search algorithm for planning two manipulators motion". *Lecture Notes in Computer Science*. Lecture Notes in Computer Science (Springer-Verlag) **1416**: 105–114. doi:10.1007/3-540-64574-8_396. ISBN 3-540-64574-8.
19. ^ Haas, O.; Burnham, K.; Mills, J. (1998). "Optimization of beam orientation in radiotherapy using planar geometry". *Physics in Medicine and Biology* **43** (8): 2179–2193. doi:10.1088/0031-9155/43/8/013. PMID 9725597.
20. ^ Harris, S.; Ifeachor, E. (1998). "Automatic design of frequency sampling filters by hybrid genetic algorithm techniques". *IEEE Transactions on Signal Processing* **46** (12): 3304–3314. doi:10.1109/78.735305.
21. ^ Augugliaro, A.; Dusonchet, L.; Riva-Sanseverino, E. (1998). "Service restoration in compensated distribution networks using a hybrid genetic algorithm". *Electric Power Systems Research* **46** (1): 59–66. doi:10.1016/S0378-7796(98)00025-X.
22. ^ Wehrens, R.; Lucasius, C.; Buydens, L.; Kateman, G. (1993). "HIPS, A hybrid self-adapting expert system for nuclear magnetic resonance spectrum interpretation using genetic algorithms". *Analytica Chimica ACTA* **277** (2): 313–324. doi:10.1016/0003-2670(93)80444-P.
23. ^ França, P.; Mendes, A.; Moscato, P. (1999). *Memetic algorithms to minimize tardiness on a single machine with sequence-dependent setup times*. Proceedings of the 5th International Conference of the Decision Sciences Institute. Athens, Greece. pp. 1708–1710.
24. ^ Costa, D. (1995). "An evolutionary tabu search algorithm and the NHL scheduling problem". *Infor 33*: 161–178.
25. ^ Aickelin, U. (1998). *Nurse rostering with genetic algorithms*. Proceedings of young operational research conference 1998. Guildford, UK.

conference 1999. Guildford, UK.

26. ^ Ozcan, E. (2007). "Memes, Self-generation and Nurse Rostering". *Lecture Notes in Computer Science*. Lecture Notes in Computer Science (Springer-Verlag) **3867**: 85–104. doi:10.1007/978-3-540-77345-0_6. ISBN 978-3-540-77344-3.
27. ^ Ozcan, E.; Onbasioglu, E. (2006). "Memetic Algorithms for Parallel Code Optimization". *International Journal of Parallel Programming* **35** (1): 33–61. doi:10.1007/s10766-006-0026-x.
28. ^ Burke, E.; Smith, A. (1999). "A memetic algorithm to schedule planned maintenance for the national grid". *Journal of Experimental Algorithmics* **4** (4): 1–13. doi:10.1145/347792.347801.
29. ^ Ozcan, E.; Basaran, C. (2009). "A Case Study of Memetic Algorithms for Constraint Optimization". *Soft Computing: A Fusion of Foundations, Methodologies and Applications* **13** (8–9): 871–882. doi:10.1007/s00500-008-0354-4.
30. ^ Areibi, S., Yang, Z. (2004). "Effective memetic algorithms for VLSI design automation = genetic algorithms + local search + multi-level clustering". *Evolutionary Computation* (MIT Press) **12** (3): 327–353. doi:10.1162/1063656041774947. PMID 15355604.
31. ^ Merz, P.; Zell, A. (2002). "Clustering Gene Expression Profiles with Memetic Algorithms". *Parallel Problem Solving from Nature — PPSN VII*. Springer. pp. 811–820. doi:10.1007/3-540-45712-7_78.
32. ^ Zexuan Zhu, Y. S. Ong and M. Dash (2007). "Markov Blanket-Embedded Genetic Algorithm for Gene Selection". *Pattern Recognition* **49** (11): 3236–3248. doi:10.1016/j.patcog.2007.02.007.
33. ^ Zexuan Zhu, Y. S. Ong and M. Dash (2007). "Wrapper-Filter Feature Selection Algorithm Using A Memetic Framework". *IEEE Transactions on Systems, Man and Cybernetics - Part B* **37** (1): 70–76. doi:10.1109/TSMCB.2006.883267.
34. ^ Zexuan Zhu, Y. S. Ong and M. Zurada (2008). "Simultaneous Identification of Full Class Relevant and Partial Class Relevant Genes". *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.
35. ^ G. Karkavitsas and G. Tsihrintzis (2011). "Automatic Music Genre Classification Using Hybrid Genetic Algorithms". *Intelligent Interactive Multimedia Systems and Services* (Springer) **11**: 323–335. doi:10.1007/978-3-642-22158-3_32.

Categories: Evolutionary algorithms

This page was last modified on 15 July 2015, at 09:49.