

Review of Clonal Selection Algorithms for Optimization

Artificial immune system (AIS) has emerged as a biologically-inspired approach that imitated the human immune systems for solving various types of computational problems such as optimization, classification and a large variety of real-world applications. [7, 24] Over the last decade of the development, the research of AIS has been divided into three types of models: clonal selection, negative selection theories, and immune networks. [19, 13]

A large proportion of studies in AIS have been focused on clonal selection algorithm which could be utilized as efficient algorithm for optimization problems. [20] Although initially thought by someone as genetic algorithm (GA) without crossover [14], clonal selection algorithm, with the features of affinity proportional reproduction and hypermutation, has been seen as a robust algorithm in the research field of Evolutionary Algorithm (EA) alongside other approaches, such as genetic algorithm and swarm intelligence algorithm. [8, 6]

The clonal selection theory [2] in an immune system describes the phenomenon that the immune system performs a natural response when binding an antigenic stimulus, where the B cells are able to recognize the antigens, and start to proliferate to provide solution to the antigens. Several types of algorithms such as CLONal selection ALgorithm (CLONALG) [6] and optimization Immune Algorithm (opt-IA) [5], have been proposed to tackle the optimization problems using the basic processes involved in clonal selection.

The basic idea of Clonal Selection Algorithm (CSA) involves two populations: a population of antigens, and a population of antibodies, where the antigens represent the problems to be solved, and the antibodies are the current candidate solutions. A basic process of CSA starts with a randomly initialization of the population of individuals (M). The affinity (fitness function value) of all antibodies (individuals) in population M are determined with respect to the antigens (the given objective function). The cloning operator will then select n best individuals with highest affinity from population M and generate n copies of these individuals proportionally to their affinity with the antigen, forming the clone population P^{clo} . The higher the affinity, the higher the number of

P^{clo} , and vice-versa. Then the hypermutation operator performs mutation to all these n individual in P^{clo} with a rate inversely proportional to their fitness values, generating the P^{hyp} . After computing the affinity of the antigen, CLONALG randomly creates new antibodies that replace the antibodies with lowest fitness in the current population. Afterwards, the algorithm repeat these process until a predefined generations are reached.

A great deal of research aiming at improving the performance of CSA has been made over the last decade. Several parameters in CSA are required to define manually: antibody population size, memory pool size, selection pool size, remainder replacement size, clonal factor, number of generations, and the random number generator seed. To make the search process much more automatically, an adaptive CSA (Adaptive Clonal Selection) is developed as an parameter version, which is tested on real-valued function optimization. [10] The ability of local search also attracts many research studies. For example, Lamarckian learning theory are introduced to enhance the local search of CSA in the Lamarckian Clonal Selection Algorithm in which recombination operator is also utilized to provide enough diversity for antibody population. [22, 11] Like this study, adding learning to the process are very common approaches to assist CSA. Baldwinian Clonal Selection Algorithm is another CSA-based algorithm developed alongside a learning theory. It takes advantage of the Baldwin effect in immune system to employ information between individual (antibodies) such that the search could be better directed. It differs from Lamarckian learning theory in that the use of the exploration performed by the antigens could lead to a better guidance in the search space. [12]

Potential improvements of the existing operator in the basic algorithm are also a major focus since the inception of CSA, especially as an powerful optimization approach. Since significance of the mutation operator in CSA which does not posses a crossover operator and relies exclusively on the mutation operator to generate new antigens, a diversity of modifications of mutation strategy has been proposed. An idea to solve complex problem is employing more mutation operators. The first thought is to implement different mutation strategies consecutively. This idea is investigated as a new approach to solve hybrid flow shop scheduling problems, in which two phased mutation procedure is implemented. [9] The generated clones undergo an inverse mutation procedure at first, then pairwise interchange mutation method is applied if the result is not favourable in the first phase. Gaussian mutation strategy, charactered with the capacity of exploitation in the local neighborhood, is introduced in another proposed algorithm for real-valued function optimization, together with a rank based selection. [3] Cauchy mutation is used in Improved CSA (IMCSA) in order to avoid premature convergence and exhibits ability of performing fast in search of the solution for job shop scheduling problem. [16] This idea is then further extended in the Fast clonal algorithm [15] that borrows the idea from fast evolutionary programming [23], in which a parallel mutation operator comprising of Gaussian and Cauchy mutation strategy are incorporated

to present an adaptive search. A chaos generator are employed to allocate both mutations aforementioned dynamically. The Cauchy mutation strategy are able to make large jumps in the search space, able to prevent the search falling into local optimum, while the Gaussian mutation shows higher probability in searching local neighbourhood, providing fine tuning ability in search of the global optima. Likewise, another study on CLONALG for constrained optimization also consider Gaussian and Cauchy random distribution as a helpful mutation scheme that make the search more efficient. [4] Furthermore, three coding schemes are investigated in the study, including binary, gray coding as well as real-valued version. Another mutation scheme is proposed in a novel CLONALG paradigm called Artificial Immune System with Mutation Multiplicity (AISMM) where multiple mutation operators are employed simultaneously to take advantage of the information gained over a number of previous generations. The fitness gain achieved in every generation is stored and used in the selection step to determine the operator selection probabilities. [1] A CSA with binary flip mutation is implemented to solve economic load dispatch problem. The mutation rate is inversely proportional to the fitness value, with the probability of mutation varying from 0.035 to 0.010. [18]

It is also worthwhile taking into account of combining mutation and other types of operators together such that the modification of genes is in accordance with the information gained in previous generations. Like in other evolutionary algorithms, a deterministic approach was first proposed to adjust the selection of antibodies, individuals to survive and to proliferate for creating the offspring generation. However, it is obvious that those antibodies selected by a deterministic selection operator are only those who exhibited best performance in the previous iteration, which could result in the search space falling into a relative small area and lead to a premature convergence. In light of the idea to overcome this drawback, research in CSA has been gradually turning into other thoughts of selection, especially a roulette wheel based selection mechanism. This type of selection mechanism provides helpful information in assisting the procedure of mutation while maintaining the diversity of antibodies to avoid premature convergence. A special version of roulette wheel selection in [9] is proposed in cloning process followed by a two phased mutation procedure. Another idea is considered to apply a super mutation operator as well as a vaccination operator to modify each individual so as to maintain the diversity of antibodies. [17] Another method proposed to extend the conventional CSA is the “psychoclonal algorithm” algorithm [21] in which Maslow’s need hierarchy theory is implemented in to solve the assembly-planning problem. Needs for mutation operator and are categorized into different level, alongside the employment of immune memory, and affinity maturation, such that the solution are better guided to the global minimum rather than local ones while also preserve the ability to remove infeasible solutions.

References

- [1] Adnan Acan. Clonal selection algorithm with operator multiplicity. In *Evolutionary Computation, 2004. CEC2004. Congress on*, volume 2, pages 1909 – 1915 Vol.2, june 2004.
- [2] F. M. Burnet. *The clonal selection theory of acquired immunity*. Vanderbilt University Press, 1959.
- [3] F. Campelo, F.G. Guimaraes, H. Igarashi, and J.A. Ramirez. A clonal selection algorithm for optimization in electromagnetics. *IEEE Transactions on Magnetics*, 41(5):1736 – 1739, may 2005.
- [4] Nareli Cruz-Cortés, Daniel Trejo-Pérez, and Carlos A. Coello Coello. Handling constraints in global optimization using an artificial immune system. In *Proceedings of the 4th international conference on Artificial Immune Systems, ICARIS'05*, pages 234–247, Berlin, Heidelberg, 2005. Springer-Verlag.
- [5] Vincenzo Cutello, Giuseppe Nicosia, and Mario Pavone. Exploring the capability of immune algorithms: A characterization of hypermutation operators. In *Proceedings of the 2004 International Conferences on Artificial Immune Systems*, pages 263–276, 2004.
- [6] L. N. de Castro and F. J. Von Zuben. Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 6(3):239–251, June 2002.
- [7] Leandro N De Castro and Jonathan Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer, 2002.
- [8] Leandro N. de Castro and Fernando J. Von Zuben. The clonal selection algorithm with engineering applications. In *Proceedings of GECCO'00, Workshop on Artificial Immune Systems and Their Applications*, July 2000.
- [9] Orhan Engin and Alper Döyen. A new approach to solve hybrid flow shop scheduling problems by artificial immune system. *Future Generation Computing System*, 20(6):1083–1095, August 2004.
- [10] Simon M. Garrett. Parameter-free, adaptive clonal selection. In *Congress on Evolutionary Computing (CEC 2004)*, volume 1. IEEE Press, 2004.
- [11] Maoguo Gong, Licheng Jiao, Jie Yang, and Fang Liu. Lamarckian learning in clonal selection algorithm for numerical optimization. *International Journal on Artificial Intelligence Tools*, 19(1):19–37, 2010.

- [12] Maoguo Gong, Licheng Jiao, and Lining Zhang. Baldwinian learning in clonal selection algorithm for optimization. *Information Science*, 180(8):1218–1236, April 2010.
- [13] Emma Hart and Jon Timmis. Application areas of ais: The past, the present and the future. *Applied Soft Computing*, 8(1):191–201, 2008.
- [14] Steven A. Hofmeyr and Stephanie A. Forrest. Architecture for an artificial immune system. *Evolutionary Computation*, 8(4):443–473, December 2000.
- [15] Nitesh Khilwani, Anoop Prakash, Ravi Shankar, and M. K. Tiwari. Fast clonal algorithm. *Engineering Applications of Artificial Intelligence*, 21(1):106–128, February 2008.
- [16] Hong Lu and Jing Yang. An improved clonal selection algorithm for job shop scheduling. In *Proceedings of the 2009 International Symposium on Intelligent Ubiquitous Computing and Education*, IUCE '09, pages 34–37, Washington, DC, USA, 2009. IEEE Computer Society.
- [17] Jia Ma, Liquan Gao, and Gang Shi. An improved immune clonal selection algorithm and its applications for vrp. In *Automation and Logistics, 2009. ICAL '09. IEEE International Conference on*, pages 2097–2100, aug. 2009.
- [18] B.K. Panigrahi, Salik R. Yadav, Shubham Agrawal, and M.K. Tiwari. A clonal algorithm to solve economic load dispatch. *Electric Power Systems Research*, 77(10):1381 – 1389, 2007.
- [19] J. Timmis, A. Hone, T. Stibor, and E. Clark. Theoretical advances in artificial immune systems. *Theoretical Computer Science*, 403(1):11–32, August 2008.
- [20] Jon Timmis. Artificial immune systems—today and tomorrow. *Natural Computing*, 6(1):1–18, March 2007.
- [21] M. Tiwari, Prakash, A. Kumar, and A. Mileham. Determination of an optimal assembly sequence using the psychoclonal algorithm. *Proceedings of the IMECH E Part B Journal of Engineering Manufacture*, 219(1):137+.
- [22] Jie Yang, Maoguo Gong, Licheng Jiao, and Lining Zhang. Improved clonal selection algorithm based on lamarckian local search technique. In *IEEE Congress on Evolutionary Computation*, pages 535–541, 2008.
- [23] Xin Yao, Yong Liu, , and Guangming Lin. Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3:82–102, 1999.

- [24] Jieqiong Zheng, Yunfang Chen, and Wei Zhang. A survey of artificial immune applications. *Artificial Intelligence Review*, 34(1):19–34, June 2010.