

COMBINED ARCHITECTURE OF GENETIC DATA AND SIMULATED ANNEALING

¹Yesvant Sree Ram M., ²Prashanth M., ³Vishnukumar K.

^{1,2,3} Electronics and Communication Engineering, Sri Krishna College of Technology, Kuniyamuthur, Coimbatore

Abstract—This paper discusses novel dedicated hardware architecture for hybrid optimization based on Genetic algorithm (GA) and Simulated Annealing (SA). The proposed architecture achieves high speed processing. Moreover, it achieves the searching not only globally, but also locally. To keep general purpose, self-control processing by a handshake system is introduced. By adopting the handshake system, the proposed architecture can be applied to various combinatorial optimization problems by only changing an encoder, a decoder, and an evaluation circuit. Furthermore, the proposed architecture realizes flexibility for many genetic operations on GA. In order to evaluate the proposed architecture, we conduct two kinds of experiments. One is an experiment which applies the proposed architecture to TSP, and the other is an experiment which applies it to VLSI floor planning. These experiment results prove that the proposed architecture achieves high speed processing, while keeping the quality of the solutions.

Keywords - Genetic algorithm, Simulated annealing, dedicated hardware, General-purpose properties

I. Introduction

Genetic Algorithm (GA)[1] was proposed by Holland as an algorithm for probabilistic search, learning, and optimization, and is based in part on the mechanism of biological evolution and Darwin's theory of evolution. This algorithm is a powerful search tool, particularly when applied for combinatorial optimization problems[2]-[7]. However, the implementation of an efficient GA often faces two major problems, on one side, the premature convergence to local optima and on the other the requirements for the GA search of long times in order to reach an optimal or a good suboptimal solution.

In order to prevent the premature convergence, the coupling of GA and one point search algorithm (local search algorithm), such as Simulated Annealing (SA)[8]-[11], to form hybrid GA can be advantageous. SA repeatedly generates succeeding solutions using the local search procedure. Some of them are accepted and some will be rejected, according to a predefined acceptance rule. The acceptance rule is motivated by an analogy with annealing processes in metallurgy as shown in Fig.1 (a). On the other hand, GA repeatedly propagates generations for a large population by applying three operators, which consist of selection, crossover and mutation as shown in Fig.1 (b). Therefore, GA requires the long computational time. The dedicated hardware for genetic algorithm processing is an important in order to reduce processing time.

In this paper, we propose the new dedicated hardware architecture for hybrid optimization based on GA and SA. It achieves the searching not only globally, but also locally. To keep general purpose, self-control processing by a handshake system is introduced. By adopting the handshake system, the proposed hardware can be applied to various combinatorial optimization problems by only

changing an encoder, a decoder, and an evaluation circuit. Furthermore, the proposed hardware realizes flexibility for many genetic operations on GA. In order to evaluate the proposed hardware, we conduct two kinds of experiments. One is an experiment that applies the proposed hardware to TSP, and the other is an experiment that applies it to VLSI floor planning. These experiment results prove that the proposed hardware achieves high speed processing, while keeping the quality of solutions.

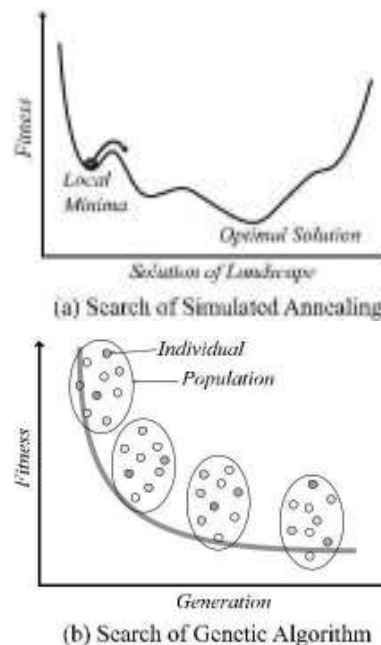


Fig.1 One point search and population based search

II. Preliminaries

A. Traveling Salesman Problem and coding

Traveling Salesman Problem (TSP)[12],[13] is one of the most basic combinatorial optimization problems. Regarding TSP, given a set of n cities and the travel cost between each pair of cities (usually distance), the salesman is to visit each once, and finally return to the start city. Therefore, a solution is an order of visiting the cities. Fig.2 shows an example of coding in TSP.

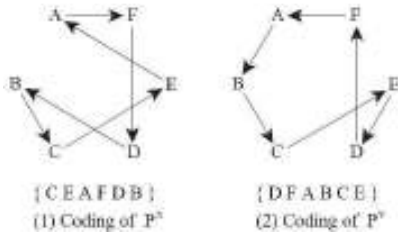


Fig.2 Example of coding in TSP

Here, if one-point crossover and multi-point crossover are adopted in this coding, it generates many lethal genes. Fig.3 shows an example of the one-point crossover procedure and the offspring generated by the crossover.

In relation to one offspring (C^x), genes of “C” and “E” are overlapped, and conversely genes of “D” and “F” are lacked in this example. The overlap and lack indicate lethal genes. The other offspring (C^y) is also the lethal genes as well as C^x . In case of C^y , genes of “D” and “F” are overlapped, and conversely genes of “C” and “E” are lacked. Therefore, several crossover operators were proposed in order to prevent generating lethal genes.

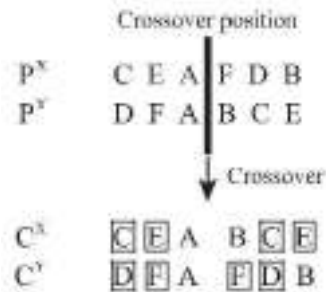


Fig.3 Example of one-point crossover

Grefenstette *et al.* [14] proposed a coding technique using a visiting city list which is created as preprocessing. An individual is represented by an order of a list instead of city name. Fig.4 shows an example of an encode procedure of Grefenstette’s approach, before crossover operation. And then, Fig.5 shows an example of one-point crossover using the encoded chromosomes. Thus, it can prevent generating lethal genes if one-point crossover and multi-point crossover are adopted. A decode procedure also requires after crossover operation. Fig.6 shows an example of the decode procedure.

The acceptance rule is motivated by an analogy with annealing processes in metallurgy as shown in Fig.1 (a).On

the other hand, GA repeatedly propagates generations for a large population by applying three operators, which consist of selection, crossover and mutation as shown in Fig.1 (b). Therefore, GA requires the long computational time. The dedicated hardware for genetic algorithm processing is an important in order to reduce processing time. Regarding TSP, given a set of n cities and the travel cost between each pair of cities (usually distance), the salesman is to visit each once, and finally return to the start city. Therefore, a solution is an order of visiting the cities. Fig.2 shows an example of coding in TSP.

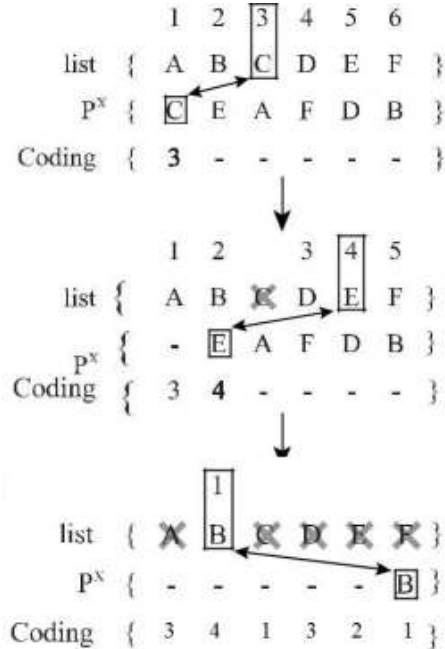


Fig.4 Example of preprocessing of Grefenstette’s approach



Fig.5 Example of one-point crossover using encoded chromosomes

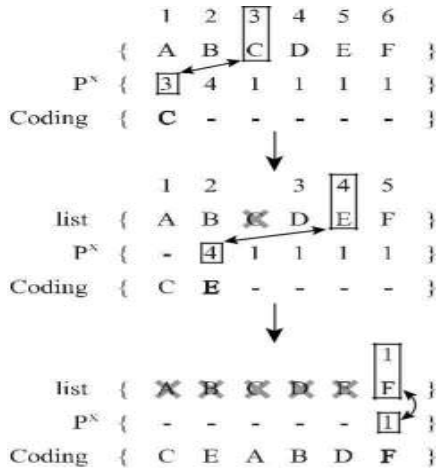


Fig.6 Example of decode procedure

B.VLSI floorplanning problem

Floorplanning[3],[15] is a generalization of the placement problem in VLSI building block layout, and it determines the coarse placement for the given set of modules. The objective of floorplanning is that (1) no modules overlap, (2) the packing area and interconnection costs are minimized. The sequence pair [15] was proposed as a representation method of floorplanning, achieving a measure success.

A sequence pair is an ordered pair of Γ^+ and Γ^- , where each of Γ^+ and Γ^- is a permutation of the names of given n blocks. Fig.7 shows a floorplan and a relative position of each block of one. Given (Γ^+, Γ^-) , one the optimal packing under the constraint can be obtained by applying the well known “longest path algorithm” for vertex weighted directed acyclic graphs (horizontal and vertical constraint graph) as shown in Fig.8.

C. Previous work

Examples of dedicated GA hardware to reduce processing time have been reported by Scott *et al.*[16], Graham *et al.*[17], and Imai *et al.*[18]. Scott *et al.* developed a hardware-based GA and demonstrated its superiority to software in speed and solution quality. Imai *et al.* proposed a processor element constituting parallel GA, and achieved the parallelism due to the number effect.

However, most of these previous works deal with small-scaled problems and limit to some fixed genetic operations. Thus, no studies have been performed to develop a dedicated hardware to realize hybrid searches that combine GA with SA.

III. Architecture Methodology

The proposed hardware has several merits: (1) hybrid optimization of GA and SA, (2) realization of high general-purpose properties, (3) introduction of a hardware-oriented hybrid algorithm, and (4) adoption of an architecture combining GA and SA.

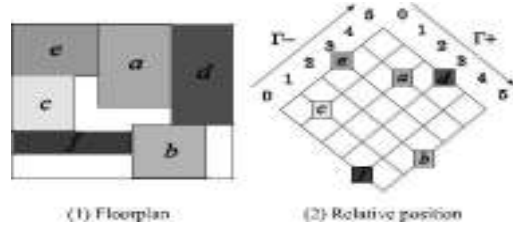


Fig.7 Example of a floorplan and a relative position

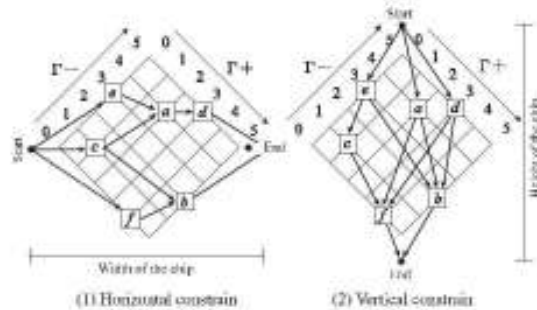


Fig.8 Examples of constrain graphs

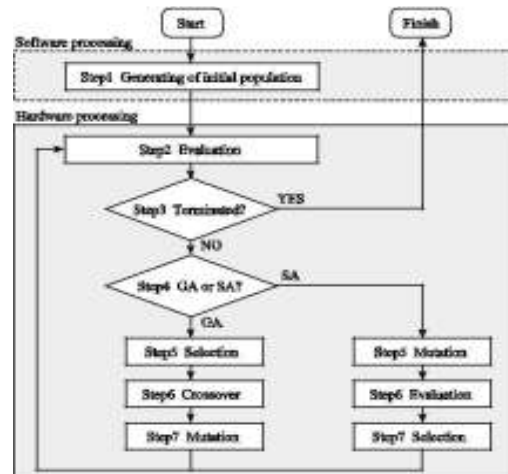


Fig.9 Processing flow of the proposed hardware

VII. Experiment And Discussion

To evaluate the proposed hardware, we applied it to TSP problem and VLSI floorplanning problem. The proposed hardware has been designed by Verilog-HDL and synthesized by the Synplicity/Synplify. The clock frequency of the proposed hardware is set up with 20 MHz. Table.1 shows the gate size. EBS indicates memory blocks of FPGA in Table.1.

We simulate the performance of the proposed hardware with the number of steps and compare it with software

processing. The experimental data of TSP problem is TSP.LIB benchmark data (eil51:51 cities), and that of VLSI floorplanning is original data(50 blocks).

The software processing was implemented in the C language and run on a Linux PC. The clock frequency of the Linux PC is 2.4GHz. In these experiments, both the software processing and the hardware processing run until the 2000th loops in SA and the 20000th generation in GA.

Table.2 shows the comparison results. In TSP problem, the proposed hardware improved the speed of 22% in comparison with software processing. Moreover, in VLSI floorplanning problem, it achieved seven times speed-up compared with software processing. The clock frequency is 20MHz for a limit of FPGA in this research. However, the proposed hardware is speeded up more, if it is implemented on LSI (Large Scale Integration).

VIII. Conclusion

In this paper, we proposed the new hardware architecture for hybrid optimization based on Genetic Algorithm and Simulated Annealing. The hybrid optimization of GA and SA in the proposed hardware achieved searching not only globally but also locally. To keep general purpose, the proposed hardware realized flexibility for many genetic operations on GA. Moreover, the gate size of the proposed hardware was reduced by adopting the combine architecture of GA and as SA. Simulation results proved that the proposed hardware achieved high speed processing as compared with software processing.

References

- [1] J.Holland, *Adaptation in Natural Artificial Systems*, the University of Michigan Press (Second edition ; MIT Press), (1992).
- [2] M.Yoshikawa, T. Fujino, and H.Terai, "A Novel Genetic Algorithm Routing Technique in 3-Dimentional Space", *Proceedings of the 10th World Multiconference on Systemics, Cybernetics and Informatics*, Vol.1, pp.70-75, 2006.
- [3] S.Nakaya, Koide and T. Wakabayashi, S, "An adaptive genetic algorithm for VLSI floor planning based on sequence-pair", *Proc. IEEE ISCAS*, Vol.3, pp.65-68, 2000.
- [4] M.Yoshikawa and H.Terai, "Asynchronous Parallel Genetic Algorithm for Congestion-Driven Placement Technique", *Proceedings of 3rd International Conference on Software Engineering Research, Management and Applications*, pp.130-136, 2005.
- [5] K.Y.Lee and P.S.Mohamed, "A real-coded genetic algorithm involving a hybrid crossover method for power plant control system design", *Proceedings of the 2002 Congress on Evolutionary Computation*, Volume 2, pp.1069-1074, 2002.
- [6] L.Zhang, Raut, R., Ling Wang, and Yingtao Jiang, "Analog Module Placement Realizing Symmetry Constraints Based on a Radiation Decoder", *Proceedings of IEEE 47th Midwest Symposium on Circuits and Systems*, pp. I-481-484, 2004.
- [7] H.Kanoh and T.Nakamura, "Knowledge based genetic algorithm for dynamic route selection", *Proceedings of Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, Vol.2, pp.616-619, 2000.
- [8] N.Metropolis, and et al., "Equation of state calculations by fast computing machines", *Journal of Chemistry and physics*, pp.1087-1092, 1953.
- [9] V.Ravi, B.S.N.Murty, and J.Reddy, "Nonequilibrium simulated-annealing algorithm applied to reliability optimization of complex systems", *IEEE Transactions on Reliability*, Volume 46, Issue 2, pp.233-239, 1997.
- [10] K.Kurbel, B.Schneider, and K.Singh, "Solving optimization problems by parallel recombinative simulated annealing on a parallel computer-an application to standard cell placement in VLSI design", *IEEE Transactions on Systems, Man and Cybernetics, Part B*, Volume 28, Issue 3, pp.454 - 461, 1998.
- [11] Lipo Wang, Sa Li, F.Tian, and Xiuju Fu, "A noisy chaotic neural network for solving combinatorial optimization problems: stochastic chaotic simulated annealing", *IEEE Transactions on Systems, Man and Cybernetics, Part B*, Volume 34, Issue 5, pp.2119 - 2125, 2004.
- [12] J.W.Pepper, B.L.Golden, and E.A.Wasil : Solving the traveling salesman problem with annealing-based heuristics: a computational study, *IEEE Transactions on Systems, Man and Cybernetics, Part A*, Vol.32, No.1, pp.72-77, 2002.
- [13] R.Baraglia, J.I.Hidalgo, and R.Perego : A hybrid heuristic for the traveling salesman problem, *IEEE Transactions on Evolutionary Computation*, Vol.5, Vo.6, pp.613-622, 2001.
- [14] J.Grefenstette, R.Gopal, B.Rosmaita, and D.VanGucht : Genetic Algorithm for the Traveling Salesman Problem, *Proc. of 1st Int. Conf. on Genetic Algorithms and their applications*, pp.160-168, 1985.
- [15] Murata,H. et al., "VLSI module placement based on rectangle-packing by the sequence-pair", *IEEE Trans. CAD*, Vol.15, No.12, pp.1518-1524, 1996.

- [16] S.D.Scott, A.Samal and S.Seth, "HGA:A Hardware-Based Genetic Algorithm", Proceedings of International Symposium on Field-Programmable Gate Array, pp.53-59, 1995.
- [17] P.Graham and B.Nelson, "Genetic Algorithm in Software and in Hardware A performance Analysis of Workstation and custom Computing Machine Implementations", FPGAs for Custom Computing Machines, pp.216-225, 1996.
- [18] T.Imai, M.Yoshikawa, H.Terai, and H.Yamauchi, "Scalable GA-Processor Architecture and Its Implementation of Processor Element", Proc. of IEEE, International Conference on Acoustics, Speech, and Signal Processing, Vol.3, pp.3148-3151, 2002.
- [19] M.Alabau, L.Idoumghar, R.Schott, "New hybrid genetic algorithms for the frequency assignment problem", IEEE Transactions on Broadcasting, Vol.48, No.1, pp.27-34, 2002.
- [20] Wang Bo, Wu Jie, Yang Jinming, and Zhao Shiwei, "A distributed hybrid wind-solar power control system based on genetic algorithm and hierarchical fuzzy control", Proceedings of Fifth World Congress on Intelligent Control and Automation, Volume 6, pp.5185-5188, 2004.
- [21] Sang Yong Yang, Lac-Jeong Park, CheolHoon Park, and Jung Woong Ra, "A hybrid algorithm using genetic algorithm and gradient-based algorithm for iterative microwave inverse scattering", Proceedings of IEEE International Conference on Evolutionary Computation, Volume 1, p.450, 1995.
- [22] M.Celino, P.Palazzari, N.Pucello, M.Rosati, and V.Rosato, "New parallel hybrid genetic algorithm based on molecular dynamics approach for energy minimization of atomistic systems", Proceedings of IEEE International Conference on Evolutionary Computation, pp.115-119, 1997.
- [23] Cheon-Seok Park and Bong-SikJeong, "Reconstruction of a high contrast and large object by using the hybrid algorithm combining a Levenberg-Marquardt algorithm and a genetic algorithm", IEEE Transactions on Magnetics, Volume 35, Issue 3, Part 1, pp.1582-1585, 1999.
- [24] Liu Jinlan, and Jian Ni, "A Hybrid Immune Genetic Algorithm Approach to Optimize the Integrated Forward/Reverse Logistics Network for 3PLs", Proceedings of Third International Conference on Natural Computation, Volume 3, pp.609-613, 2007.
- [25] Jingjing Wu, KelinXu, Qinghua Kong, and Wenxian Jiang, "Application of the Hybrid Genetic Algorithm to Combinatorial Optimization Problems in Flow-shop Scheduling", Proceedings of International Conference on Mechatronics and Automation, pp. 1272-1277, 2007.
- [26] Aihua Wang, and Weizi Yang, "Design of Energy Management Strategy in Hybrid Electric Vehicles by Evolutionary Fuzzy System Part II: Tuning Fuzzy Controller by Genetic Algorithms", Proceedings of The Sixth World Congress on Intelligent Control and Automation, Volume 2, pp.8329-8333, 2006.
- [27] Decai Huang, HaidongGuo, and NengQian, "Hybrid GeneticAlgorithm for Minimizing the Range of Lateness and Make-span on Non-identical Parallel Machines", Proceedings of InternationalConference on Neural Networks and Brain, Volume 1, pp.150-154, 2005.
- [28] K.P.Wong, and Y.W.Wong, "Hybrid genetic/simulated annealing approach to short-term multiple-fuel-constrained generation scheduling", IEEE Transactions on Power Systems, Volume 12, Issue.