Available online at www.elixirpublishers.com (Elixir International Journal)

Electrical Engineering

Elixir Elec. Engg. 99 (2016) 43029-43031

ISSN: 2229-712X

Codesing Partitioning Using Memetic Algorithm in VLSI: A Review

S. Prakasam*, M. Venkatachalam, M. Saroja and N. Pradheep

Department of Electronics, Erode Arts and Science College (Autonoous), Erode-9, Tamilnadu, India.

ARTICLE INFO

Article history:

Received: 24 August 2016;

Received in revised form:

27 September 2016;

Accepted: 6 October 2016;

Keywords

Codesign partitioning,

Memetic algorithm,

VLSI design.

ABSTRACT

The complexity and size of circuits have been rapidly rising, placing a stressing demand on industry for faster and more competent for VLSI design. If partitioning is not done in valuable manner, ignoring the parameters like firmness, time delay and robustness it may corrupt the overall performance of a design. Optimization is used to make a design particularly efficient, finding the maximum of a function. In the partitioning main objective is to minimize the number of cuts. Investigating the application of the Memetic Algorithm (MA) for solving the codesign partitioning problem can be done.

Introduction

The process of creating Integrated Circuits (IC) by merging thousands of transistors into a single chip began in the 1970s and it called as Very Large Scale Integration. VLSI technology is moving towards miniaturization. In circuit petitioning, the circuit is separated into bi-partitioning and multi-way partitioning. Partitioning is applied recursively in large circuits until the complexity in each part is compact to extend that it can be handled efficiently with existing tools [1]. Optimization is a procedure to find an alternative with the most cost effective or highest achievable performance by improving the desired factors and reducing the undesired ones. Important considerations include the minimum area of each partition, minimum number of interconnection i.e., mini cut, logic functionality, delay due to partitioning and fitness function which is determined to improve the circuit parameters to obtain good performance. Next comes the available algorithm technology, which concludes how effectively we can deliver a given partitioning formulation and optimize a given objective [2]. Since the exact algorithm is often slow when applied to practical problems, heuristic and meta-heuristic approaches are usually ideal solution methods. In different researches have achieved varying levels of achievement using various optimization techniques. Hybridization of evolutionary algorithms with local search has been investigated in many studies.

In modern years, there has been an growing interest in the study of the codesign partitioning problem. Several approaches to the problem have been developed. However, since most of the exact formulations of the partitioning problem are NP-hard, the practical usefulness of these approaches is limited to fairly small problem instances only. For larger instances, a number of heuristic algorithms, both conventional and general purposes, have been proposed. Up to now, many general-purpose heuristics or meta-heuristics have also been applied to solve the partitioning problem. It must be mentioned that's the codesign partitioning problem is to minimize a cost while satisfying some design constraints. In fact, the problem is a discrete inhibited optimization problem, and the optimal solutions usually lie on the boundary of the feasible region [3].

Hence, it is necessary to develop effective techniques in handling the constraints. A memetic based algorithm can solve the codesign partitioning problem.

The motivation behind this research is to improve performance of a memetic algorithm [4] for VLSI circuit partitioning by introducing a hardware design that exploits the inherent parallelism of the algorithm. The final goal is to develop a hardware implementation of a Memetic algorithm by incorporating a Local Search into the design to produce better results.

Literature Survey

Genetic algorithm for circuit partitioning has been attempted by Roy *et al.*, (2012) [5]. In this practical paper solution is easy and we can easily apply genetic operator in this type of problem. Complexity is measured in both time and space, provided size of problem and as integer (count is infinite). They show multi way circuit partition using genetic algorithm and the main advantage of the circuit partitioning using genetics algorithm is "we can easily multi way partitioning in many types of VLSI circuit". But time is not reduced it only find minimum cut size. Hardware/software partitioning is one of the crucial steps of codesign systems. Most formulations of hardware/software partitioning problem are NP-hard.

Lin *et al.*, (2012) [6] investigate the application of the memetic algorithm for solving the hardware/software partitioning problem on an NP-hard model. The memetic algorithm uses a local search procedure, which is based on simulated annealing, to improve initial individuals. It is used for solving NP hard problem which uses a local search procedure to improve initial individuals. The cost for implementing this memetic algorithm is high.

Lin *et al.*, (2014) [7] propose a tabu search-based memetic algorithm to solve the HW/SW partitioning problem. The algorithm uses a tabu search as its local search procedure. This tabu search has a special feature with respect to solution generation, and it uses a feedback mechanism for updating the tabu tenure. Hence the memetic algorithm is applied to this unconstrained problem, and robustness with respect to the quality of optimal solutions. The drawback is execution cost is high

Tele:

E-mail address: prakash.sasc@gmail.com

Reconfigurable system on chip is well known for its flexibility for high performance embedded systems [8]. Du *et al.,* (2014) apply the SFLA algorithm to solving hardware/software partitioning problem on reconfigurable system on chip with coarse-grained. A benefit is that the potential of SFLA algorithm is substantially exploited so that we can get a short finishing time cost of the critical path. But in the future this SFLA algorithm can be improved and to be specific.

Bahri *et al.,* (2013) [9] aims to provide Hardware/Software (Hw/Sw) codesign guidelines for system-on-chip field-programmable gate array-based sensor less ac drive applications. This Hw/Sw partitioning is performed taking into account both the control requirements (bandwidth and stability margin) and the architectural constraints (e.g., available area, memory, and hardware multipliers). The proposed algorithm is an efficient to search for optimum solutions very fast and be used for functional partitioning of larger embedded systems. But the drawback is it should not find increasing attention and applications

A heuristic solution is proposed by Han *et al.,* (2013) [10] for scheduling and partitioning on multi-processor system on chips (MPSOC). In order to minimize the overall execution time, the proposed algorithm assigns different priorities to different tasks according to their out-degree and the software execution time. The scheduling algorithm can reduce unnecessary waiting time by assigning different priorities to different tasks on the basis of their out-degree and software execution time, so as to minimize the overall execution time. But this work can be extended to reduce the cost.

Many types of embedded systems applications are implemented as a combination of software and hardware. For such systems the mapping of the application units into hardware and software, i.e. the partitioning process, is a key phase of the design. Although there exist techniques for partitioning, the entire process, in particular in relation to different application requirements and project constraints, is not properly supported. This leads to several unplanned iterations, redesigns and interruptions due to uncontrolled dependencies between hardware and software parts. In order to overcome these problems [11], Sapienza *et al.,* (2013) provide a design process that enables the partitioning based on a multiple criteria decision analysis in a late design phase. From this design of an overall process suitable for enabling (i) platform independent design and reuse, and (ii) a systematic decision process to partition applications in a late stage of the design phase. But there is need for the formalization of a meta-model for enabling an accurate modeling of hardware and software components

Hardware/software partitioning is a crucial problem in hardware/software co-design. Luo *et al.,* (2012) [12] deeply investigate genetic algorithm (GA) for hardware/software partitioning, our co-design targets a heterogeneous multicore system on chip (SoC) which consists of several different types of processing engines(PE), Communicating structure adopts NOC, used GA for four task graphs to simulate the hardware/software partitioning, experiments show our method is an effective hardware/software partitioning algorithm. This algorithm will determine the best partition that minimizes the logic area. But this work can extends the system model to encompass other quality attributes, like power consumption, influence of communications, and the degree of parallelism.

Mishra *et al.,* (2012) [13] presents a framework for identifying such functions by proposing an algorithm.

The framework uses the time profiling and clustering technique to achieve the objectives. Open source spark compiler has been used to convert functions to hardware description language. The final interfacing has been done in embedded development kit. The benefit in this process is that the design becomes faster although the complexity involved in the design requires more expertise. This work can be extended to improve the algorithm by more closely coupling compiler optimizations with the hardware/software partitioning

Pando *et al.,* (2013) [14] presents the results of applying a fuzzy approach to the HSP problem. This approach is more flexible than many others due to the fact that it is possible to accept quite good solutions or to reject other ones which do not seem good. The advantage of this analysis facilitates decision making in selecting the most appropriate algorithm depending on the application constraints but does not give solutions in the Pareto front.

In the partitioning main objective is to minimize the number of cuts. For this, PSO algorithm is proposed is Singh *et al.,* (2013) [15] for the optimization of VLSI interconnection (net list) bipartition. Meanwhile, the corresponding evaluation function and the operators of crossover and mutation are designed. The algorithm is implemented to test various benchmark circuits. It's increasing in the number of iteration we can minimize the number of cuts in the different circuit series. This work algorithm can be extended by improving the techniques which provides better solution than this algorithm.

Solanke *et al.,* (2013) [16] presents framework for Research work to analyse effectiveness of System Verilog for system level modelling and Co-design. It further discusses how system Verilog can be used for modelling software components as well as hardware components of the system. One method is also proposed to carry out the Co-simulation and Codesign. Paper also discussed the expected results from the proposed work. System Verilog, will overcome the difficulties involve in traditional hardware/software codesign method and will help Design Engineers to develop the systems more accurately with minimum efforts. But the time consuming is high.

Teich (2012) [17] presents major achievements of two decades of research on methods and tools for hardware/software codesign by starting with a historical survey of its roots, by highlighting its major research directions and achievements until today, and finally, by predicting in which direction research in codesign might evolve in the decades to come. In this work will provide the researches basic benefits and work of hardware/software design.

The above reviews show the problems and conclusions of several literatures, different authors give various ideas to solve the codesign partitioning problem. Still there is a problem includes in codesign partitioning are high cost and execution time are directed in next section.

Problems and Directions

Partitioning usually consumes much time. The difficulties of partitioning are; (1).completely determines the following stages (Hardware and Software synthesis) NP-hard problem, (2). Generally made "by hand", relying on experience. It is imperative to make it as efficient as possible to avoid costly feedbacks. Problem of VLSI circuit partitioning is non polynomial solid and cannot be effectively solved by deterministic algorithms.

The problem involves isolating the circuit net list into two subsets and some of the connections (edges) are also cut. The number of edges belonging to two different partitions (Traditional and Automatic) is the cost of a partition. Traditionally partitioning was carried out manually, causing substantial delay. Automatic partitioning between hardware and software plays a key role and the results of partitioning directly affect the system area and processing time. When building embedded systems, executing functions directly in the hardware will always provide more efficient computation because of the simple fact that it avoids the overhead that software operations provide. Hard wiring functions in a device's structure is thus preferred for its fast, however, there is a cost one has to pay: the more functions we want to apply in hardware, the bigger area of hardware needed. In real world situations, there is always a strict limit on the amount of hardware that one may use in building a single instance of an electronic device, so it is a good idea to decide on only a small subset of operations to be executed in hardware, while leaving everything else to run as software.

Conclusion

In advances technology, Computer Aided Design (CAD) tools play an important role in the VLSI physical design process. The physical design process is becoming increasingly complex allowing for more transistors to be integrated onto a single die. Although new techniques are continuously being investigated, the common goal of each algorithm is to produce better results in less time. Traditionally, these algorithms have been created in software due to its flexibility and ease of development. One of the drawbacks of software based programs, however, is that they execute in a sequential manner resulting in inefficient time usage. This has lead to the investigation of hardware algorithms to replace traditional software tools. As digital systems become more complex and performance criteria become more stringent, codesign will become a necessity. Improved design tools and unified design environments will allow codesign techniques to become standard practice. A memetic algorithm is better design for solving the codesign partitioning problem in VLSI.

References

1. Saha, D., Mitra, R.S., Basu, A.: Hardware software partitioning using genetic algorithm. In: Agrawal, V., Mahabala, H. (eds.) Proc. of the 10th Int'l Conf. on VLSI Design, pp. 155–160. IEEE Computer Society Press, Hyderabad (1997)
2. Ji, Y., Li, L.Y., Shi, M., Zhang, L.L.: Hardware/software partitioning algorithm using hybrid genetic and tabu search. Computer Engineering and Applications 45(20), 81–83 (2009) (in Chinese).
3. Wiangtong, T., Cheung, P., Luk,W.: Comparing three heuristic search methods for functional partitioning in hardware-software codesign. Journal of Design Automation for Embedded Systems 6(4), 425–449 (2002).
4. Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by simulated annealing. Science 220, 671–680 (1983).
5. Roy, Sharadindu, and Samar Sen Sarma. "Improvement Of The Quality Of Vlsi Circuit Partitioning Problem Using Genetic Algorithm‖." Journal of Global Research in Computer Science 3, no. 12 (2012).
6. Lin, Geng, Wenxing Zhu, and Jinglan Wu. "A Memetic Algorithm for Hardware Software Partitioning." In Recent Advances in Computer Science and Information Engineering, pp. 129-134. Springer Berlin Heidelberg, 2012.
7. Lin, Geng, Wenxing Zhu, and M. Montaz Ali. "A Tabu Search-Based Memetic Algorithm for Hardware/Software Partitioning." Mathematical Problems in Engineering 2014 (2014).
8. Du, Jiayi, Xiangsheng Kong, Xin Zuo, Lingyan Zhang, and Aijia Ouyang. "Shuffled Frog Leaping Algorithm for Hardware/Software Partitioning." Journal of Computers 9, no. 11 (2014): 2752-2760.
9. Bahri, Imen, Lahoucine Idkhajine, Eric Monmasson, and M. El Amine Benkhelifa. "Hardware/software codesign guidelines for system on chip FPGA-based sensorless AC drive applications." Industrial Informatics, IEEE Transactions on 9, no. 4 (2013): 2165-2176.
10. Han, Honglei, Wenju Liu, Jigang Wu, and Guiyuan Jiang. "Efficient algorithm for hardware/software partitioning and scheduling on MPSoC." Journal of Computers 8, no. 1 (2013): 61-68.
11. Sapienza, Gaetana, Tiberiu Seceleanu, and Ivica Crnknovic. "Partitioning decision process for embedded hardware and software deployment." InComputer Software and Applications Conference Workshops (COMPSACW), 2013 IEEE 37th Annual, pp. 674-680. IEEE, 2013.
12. Luo, Li, Hongjun He, Qiang Dou, and Weixia Xu. "Hardware/Software partitioning for heterogeneous multicore SoC using genetic algorithm." InIntelligent System Design and Engineering Application (ISDEA), 2012 Second International Conference on, pp. 1267-1270. IEEE, 2012.
13. Mishra, Ashish, Kritika Garg, A. R. Asati, and K. Solomon Raju. "Hardware software co-design using profiling and clustering." In Communication, Information & Computing Technology (ICCICT), 2012 International Conference on, pp. 1-4. IEEE, 2012.
14. Pando, Humberto Díaz, Sergio Cuenca Asensi, Roberto Sepúlveda Lima, J. F. Calderın, and Alejandro Rosete Suárez. "An application of fuzzy logic for hardware/software partitioning in embedded systems." Computación y Sistemas17, no. 1 (2013): 25-39.
15. Singh, Rajdeep, Kumari Kalpna, and Dawindar Kumar Mishra. "Hybrid Optimization Technique for Circuit Partitioning Using PSO and Genetic Algorithm." International Journal of Emerging Trends in Electrical and Electronics (IJETEE–ISSN: 2320-9569) 4, no. 2 (2013): 69-71.
16. Solanke, Gitanjali R., Dipti Y. Sakhare, and Manish M. Patil. "Hardware Software Partitioning For A Digital System & Its Validation Using FPGA." InInternational Journal of Engineering Research and Technology, vol. 2, no. 3 (March-2013). ESRSA Publications, 2013.
17. Teich, Jürgen. "Hardware/software codesign: The past, the present, and predicting the future." Proceedings of the IEEE 100, no. Special Centennial Issue (2012): 1411-1430.