Journal of Engg. Research **Vol. 3 No. (1) Mar 2015** pp. 79-93

خوارزمية مركبة لمعالجة العيوب الناتجة عن تطبيق تقنية النانو

حصة المطيري و امتياز أحمد

*كلية هندسة وعلوم الحاسوب، قسم هندسة الكمبيوتر، جامعة الكويت – الكويت

الخلاصـة

برزت تكنولوجيا النانو كواحدة من التقنيات البديلة الواعدة لتكنولوجيا CMOS بسبب ارتفاع كثافتها وسرعتها العالية، وخفة وزنها، وانخفاض استهلاكها للطاقة. ومع ذلك فإن الأعطال تظهر بشكل عالي في تكنولوجيا النانو. وبالتالي، فإن الحاجة إلى تقنيات للتعامل مع الأعطال تصبح بالغة الأهمية في تكنولوجيا النانو.في هذا البحث تتم معالجة مشكلة العثور على أكبر عدد من المفاتيح الخالية من الأعطال في crossbar المليئة بالأعطال من أجل عائدات أعلى. يتم اقتراح خوارزمية مركبة عن طريق دمج طرق بحثية معروفة مع الخوارزمية الجينية (GA) للبحث عن حلول لاستخراج مساحة كبيرة خالية من الأعطال. الخوارزمية المقترحة تستغل فكرة درجات العقد والتي تلعب دورا حاسما في آلية الاختيار في طرق البحث المختارة للحصول على أفضل الحلول. في الخوارزمية المقترحة، وGA تقدم ترتيب جديد للاختيار عن طريق إنتاج مجموعة من الدرجات بواسطة طرق البحث المختارة من أجل استخراج قيمة جديدة sub-crossbar الخالية من العيوب. نتائج التجارب أثبتت فعالية الخوارزمية المركبة المقترحة في ايجاد مسافة اكبر خالي من العيوب من العارضة المليئة بالعيوب مقارنة مع الطرق المستخدمة حاليًا.

A hybrid mapping algorithm for reconfigurable nano-architectures

Hessa K. Al-Mutairi and Imtiaz Ahmad

Computer Engineering Department

Kuwait University, Kuwait

Corresponding Author: Imtiaz Ahmad, Email: imtiaz.ahmad@ku.edu.kw

Phone: (+965)2498-5849, Fax: (+965)2483-9461

ABSTRACT

Nanotechnology is emerging as one of the most promising alternative technology to CMOS technology because of its higher density, high speed, lighter, and lower power consumption; however, defects are much higher in nanotechnology. Therefore, the need for defect-tolerance techniques becomes crucial in nanotechnology. This paper addresses an important intractable problem of finding a maximum size defect-free sub-crossbar in defective nano-scale crossbars for a higher yield. We propose a hybrid mapping algorithm by embedding known greedy heuristics with genetic algorithm (GA) to search a large solution space effectively. The proposed algorithm exploits the degrees of nodes, which play a crucial role in the selection mechanism in the greedy mapping heuristics to generate a better quality solution. In the proposed algorithm, GA provides the selection order by generating a new set of degrees that are used by the greedy mapping heuristic to find a new value for the defect-free sub-crossbar (k). The experimental results demonstrate the effectiveness of the proposed hybrid algorithm in finding a large size defect-free sub-crossbar compared to the existing state-of-the-art greedy heuristics.

Keywords: Biclique problem; defect tolerance; genetic algorithm (GA); mapping algorithm; nano-crossbar switches; nanotechnology

INTRODUCTION

The limitation of CMOS technology in electronic and computational circuits incites researchers to focus on finding alternative technologies such as nanotechnologies (Rao *et al.*, 2006). Unlike CMOS devices, nano-electronic devices are faster, consume less power, and occupy a smaller area because of their size and density factors (Zieglar & Stan, 2003). In nanotechnologies, 2D crossbars can be built easily because of their regular structure, which plays a key role in reconfigurable nano-architecture (Paul & Bhunia, 2012; Zamani *et al.*, 2014). Unfortunately, defects are new challenges that arise when nano-wires are integrated into computing and electronic systems. During the manufacturing process of nano-crossbars, defects are either non-programmable crosspoint defects or disconnected wire defects (Ghavami *et al.,* 2010). These defects are catastrophic; they make switches unusable and prevent them from being reconfigurable. An example of a 3x3 defective crossbar is shown in Figure 1(a). It consists of two sets of orthogonal nano-wires. The horizontal wires labeled "U" are the inputs, whereas the vertical wires labeled "V" are the outputs. At each crosspoint, there is a programmable switch. The defective switches in this crossbar are represented as "X". The need for defect-tolerance techniques becomes crucially important especially when mapping any function to a defective crossbar.

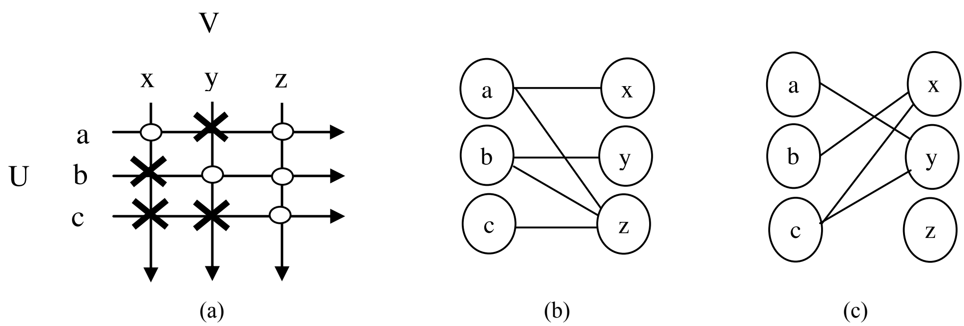

Fig. 1. (a) A 3x3 crossbar with four defects, (b) representative graph, (c) complement graph

Most researchers are exploiting fault tolerance techniques to get full utilization of defective crossbars and reaching a high manufacturing yield. Fault tolerant approaches are classified as defect-aware and defect-unaware approaches (Bonam *et al.*, 2007). In defect-aware design flow, the defect maps are very large, since they identify all the defective crosspoints. The size of the resulted map is $O(n^2)$. With this size, it is difficult to retrieve this map each time especially in online testing. A variety of techniques have been reported for defect-aware design flows to handle defect problems in nano-circuits (Zheng & Huang, 2009; Su & Rao, 2009, 2014; Yang & Datta, 2011; Gören *et al.*, 2011). However, in defect-unaware design flow, only two binary vectors are needed to be retrieved and stored with a size equal to O(n). These two vectors denote the horizontal and vertical lines in a crossbar (Tahoori, 2005). In this flow, there is no need to consider the location of defects in high level design steps. In this flow, a maximum size defect-free sub-crossbar is identified, which means that a defective crossbar can be used as a defect-free sub-crossbar with a reduced size. A defect-aware approach is required only in the final mapping step. Maximizing the value of defect-free sub-crossbar (k) is important for efficient function mapping in nano-scale crossbars and to improve the yield of a defective crossbar.

RELATED WORK

In general, most of the algorithms for identifying the maximum defect-free sub-crossbar convert the defective crossbar into a representative graph. A bipartite graph G (U, V, E) is called a representative graph of a crossbar with two partitions: U as the set of input nano-wires and V as the set of output nano-wires. E represents all edges between U and V, which in turn indicate the non-defective crosspoints of the defective crossbar. A bipartite graph is called a complete bipartite graph, if and only if, all nodes in U and V are connected to each other. The two-dimensional crossbar in Fig. 1(a) can be described by a representative graph as shown in Fig. 1(b). A bipartite graph G (U, V, E) can be called a biclique, if and only if, E = U x V in which E is the total number of possible edges in the bipartite graph. A biclique G (U, V, E) is balanced if |U|=|V|. A maximum complete subgraph of a bipartite graph is a maximum biclique (Tahoori, 2005, 2006; Al-Yamani *et al.*, 2007; Yuan & Li, 2011, 2014, Yuan *et al.*, 2014). The crossbar can still be used as a maximum k x k (k < n) sub-crossbar if a maximum biclique of size k can be extracted.

The complement graph of a graph G is a graph $\overline{G}$ with the same set of vertices such that two vertices of $\overline{G}$ are adjacent and only if they are not adjacent in G. The complement graph of the graph in Fig. 1(b) is represented in Fig. 1(c). An independent set S in a graph G is a subset of nodes that are disconnected. The maximum independent set is an independent set with the maximum number of nodes. Then, extracting the maximum biclique from a graph is the same as extracting the maximum independent set in a complement graph (Fig. 1(c)). In most of algorithms for finding the maximum independent set in a graph, the node selection order plays a key role in the final solution quality. This requires having a well-defined selection process for the nodes in the graph. Many research papers have suggested different ways for implementing the selection process. The node degree plays a significant role in the selection process. In Magun, 1998 and Langguth *et al.*, 2010, the authors compared different degree based selection heuristics such as simple greedy, static Mindegree, dynamic Mindegree, and others.

The three state-of-the-art greedy heuristics proposed in the literature (Tahoori, 2005, 2006; Al-Yamani *et al.*, 2007; Yuan & Li, 2011, 2014) to identify the maximum defect-free sub-crossbar from the defective crossbar also used degree-based selection criteria. These algorithms were designed to supercede each other with the aim of improving the time complexity and the size of the defect-free sub-crossbar. These algorithms work on extracting the maximum independent set in the complement graph. Each algorithm deletes the nodes in a particular order depending on the degrees of the nodes. The algorithm reported in Tahoori, 2005 and 2006 starts by constructing the complement graph of the representative graph with two sets, U and V. After that, it tries to find the maximum independent set in the complement graph by removing maximum-degree nodes. In order to keep the graph balanced, the algorithm alternates between U and V while deleting the maximum-degree nodes. Each time a node with zero degree will be added to the maximum independent set. The final nodes in the independent set (k) will be considered as the final defect-free subset (kxk). The time complexity of this algorithm is $O(n^2)$ in which n represents the number of nodes in the graph. The algorithm given in Al-Yamani *et al.*, 2007 also tries to find the maximum balanced independent set in the complement graph. Each time, it removes the node on one side that is connected to a maximum number of minimum-degree nodes on the other side. It alternates between U and V while deleting a node to keep the graph balanced. The time complexity of the algorithm is $O(n^3)$.

The algorithm given in Yuan & Li, 2011 follows a different scenario to find the maximum-balanced independent set. While alternating between U and V, it removes a node in one partition with the maximum degree that is connected to a minimum-degree node in the other partition. The time complexity of the algorithm is $O(n^2)$. Recently, a new evolutionary algorithm with structure mutation was proposed for the maximum balanced biclique problem (Yuan *et al.*, 2014). In the proposed algorithm, local search is complemented with a repair-assisted restart process, and a new mutation operator is used to avoid the local minima and enhance the exploration space. Since the problem of finding a maximum size, defect-free sub-crossbar is an NP-complete problem (Peeters, 2003; Shrestha *et al.*, 2011), an algorithm that finds an optimal solution in polynomial time is unlikely to exist. Therefore, optimal solution strategies must be sacrificed in favor of fast heuristic techniques.

PROPOSED HYBRID ALGORITHM

As explained earlier, all the greedy algorithms for extracting a defect-free sub-crossbar from a defective crossbar depend on the node selection order. This means that the selection order, which is based on the node degrees, is very critical in determining a better quality solution. Based on that fact, this paper will introduce a hybrid algorithm, which integrates a genetic algorithm (GA) with a known greedy heuristic. Genetic algorithms are probabilistic combinatorial optimization techniques in which genetic operators such as selection, crossover, and mutation, are derived from the selection processes in nature. They guide a population of potential solutions to a problem toward optimal solutions. GAs have been successfully applied to a wide range of problems in diverse fields (Lim, 2014).

GAs are blind search techniques and they require problem-specific genetic operators (selection, crossover, mutation) to achieve optimal solutions. The proposed hybrid method exploits a new search space (Storer *et al.,* 1992; Dhodhi *et al.*, 2002), which integrates a known fast problem specific heuristic with the local search. The key concept in this method is to base the definition of the search neighborhood on a heuristic/problem pair (h, p), where h is a known fast heuristic and p represents the problem data. Since a heuristic h is mapping from a problem to a solution, the pair (h, p) is an encoding of a specific solution. By perturbing the problem p, the neighborhood of solutions is generated. This neighborhood forms the basis for a local search. The problem space is generated by perturbing the problem data, which is the degree of a node in the problem addressed. For example, let P be a set of m problems obtained by perturbing the original problem data. That is, $P = \{p_j = p_0 + \delta, j = 1, ..., m\}$, where p_0 is the data for the original problem (degree of the node in the complement graph), and δ is the randomly generated perturbation vector. The perturbation range depends on the specific problem. In order to keep the generated "dummy" problem values in proximity of the original problem values, upper and lower limits on the perturbation can be introduced (difference between the maximum and the minimum degree). The solution subset S corresponding to the problem set P can be created by the application of an known heuristic: h, $S = \{h(p_j), j=1, ..., m\}$. In our case, any of the known state-of-the-art greedy heuristics in literature (Tahoori, 2005, 2006; Al-Yamani *et al.*, 2007; Yuan & Li, 2011) can be embedded with GA. All these heuristics are sensitive to the node degree for their selection order. The high-level description of the proposed algorithm is shown in Fig. 2.

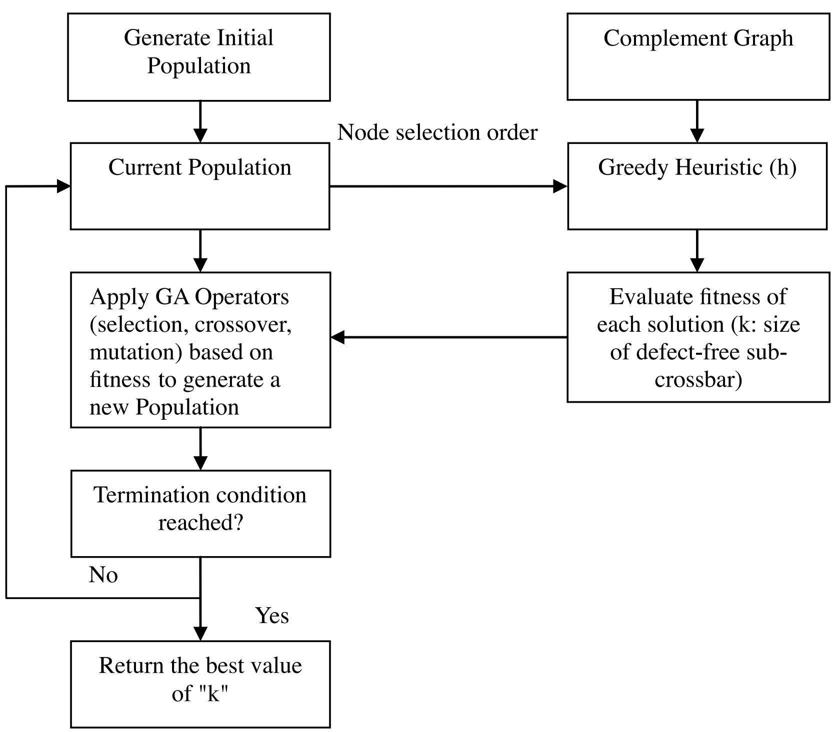

Fig. 2. The proposed hybrid algorithm

Each chromosome in the population consists of an array of integers representing the degree for each node of the graph (for both U set and V set in the complement graph). The degree of each node for the first chromosome is computed from the complement graph. The rest of the chromosomes in the initial population are generated by a random perturbation in the degree in the neighborhood. The greedy heuristic (h) was applied to generate a solution from a given chromosome with the objective of finding the value of k (size of the maximum defect-free sub-crossbar). The cost function is the key issue as it reflects the goal of the optimization. Note that the chromosome can only provide the selection order for the heuristic to select nodes from the complement graph to generate a solution. The fitness is then evaluated based on the solution from the original complement graph.

Genetic operators such as selection, crossover, and mutation are the key elements of the genetic algorithms. The selection operator chooses chromosomes for reproduction from current population based on their relative fitness. Chromosomes with higher fitness will have a higher probability of contributing one or more offspring in the next generation. The selection method was implemented using a biased roulette wheel where each chromosome in the population has a slot sized in proportion to its fitness. Each time we require an offspring, a simple spin of the weighted roulette wheel gives a parent chromosome. The crossover operator takes two parent chromosomes selected by the selection operator from the current population and generates two children by incorporating features from both parents. The premise here is that through this process desirable features are enhanced, while most undesirable features are suppressed. We have applied a one-point crossover operator to the degree of the chromosome. In the one-point crossover operator, a cross site is selected randomly, and the value of the degree to the right of the cross site is swapped among the two mating chromosomes. The crossover is applied with a certain crossover rate, which is the ratio of the number of offspring produced by crossover in each generation to the population size. It controls the amount of crossover being applied.

In nature, mutation refers to spontaneous and random changes in genes. In a genetic algorithm based approach, mutation introduces new features into the current population by altering a randomly picked gene value. Mutation was implemented by selecting a gene at random with a mutation rate and perturbing its value. The mutation rate is the percentage of the total number of genes in the population which are mutated in each generation. The objective is to explore a wider space of node degrees but within the proximity to the original problem. We also employed the selection scheme based on elitist selection to keep the best parents in the next generation. The best parents in our problem are those who gave a large value of k.

The core idea is to use GA in generating a new set of degrees each time and to use these new degrees in a greedy algorithm for the selection process. Each time, the greedy algorithm will be executed with a new set of degrees, which will result in a new solution. Only the best solution is kept among all iterations until the termination criteria is reached. The evolution process only helps us in finding the selection order, which leads to a better solution.

EXPERIMENTAL RESULTS

The proposed hybrid algorithm and the three state-of-the-art greedy algorithms proposed in literature: Algorithm 1 (Tahoori, 2005, 2006), Algorithm 2 (Al-Yamani *et al.*, 2007), and Algorithm 3 (Yuan & Li, 2011) to identify the maximum defect-free sub-crossbar from the defective crossbar, were simulated using C++ language. The experiments were performed using a Windows-7 running on a laptop with an Intel Core 2.4 GHz processor and 4GB memory. For each data point, 100 crossbars with different sizes (n) (32, 64, 250, 500 and 1000) and with defect rates (p) (5% to 30%) were randomly generated. It was determined experimentally that for the given problem population size of 20 and the number of generations of 100 were sufficient to arrive at good solutions. The size of the chromosome was the total number of nodes in the graph. It may appear that the number of generations was too low as compared with traditional GAs. One reason for arriving at good solutions earlier is the application of a problem specific heuristic (that is, a heuristic guided search instead of a blind search). The crossover rate =0.1 and mutation rate=0.001 performed quite well. By operating in problem space, the proposed algorithm had a fast convergence. The parameters of the GA were fixed while simulating the algorithms. We embedded all the three greedy algorithms with GA; hybrid algorithm 1 is a combination of algorithm 1 with GA, hybrid algorithm 2 is a combination of algorithm 2 with GA, while hybrid algorithm 3 is a combination of algorithm 3 with GA.

To measure the effectiveness of each algorithm, the average sizes of maximum defect-free sub-crossbars resulted in these algorithms with different values of n and p that were compared by computing the percentage improvement. The formula for computing the percentage improvement % is:

$$\left[\left[\frac{new\_value}{original\_value} \right] - 1 \right] \times 100 \tag{1}$$

Note that in our algorithm, since we embedded the known heuristics with GA, the first chromosome in the initial population generated the same solution as reported by these heuristics. The GA further improves the solution quality by evolving the selection order. The percentage improvement of the hybrid algorithm 1 over algorithm 1, hybrid algorithm 2 over algorithm 2 and hybrid algorithm 3 over algorithm 3 with respect to different defect densities is shown in Figures 3-5, respectively. These figures show that hybrid algorithm 1, hybrid algorithm 2 and hybrid algorithm 3 are much more effective than implementing the three greedy algorithms alone especially for large defect densities. In other words, we can utilize more potential of the given defective crossbar by combining GA with the greedy algorithms.

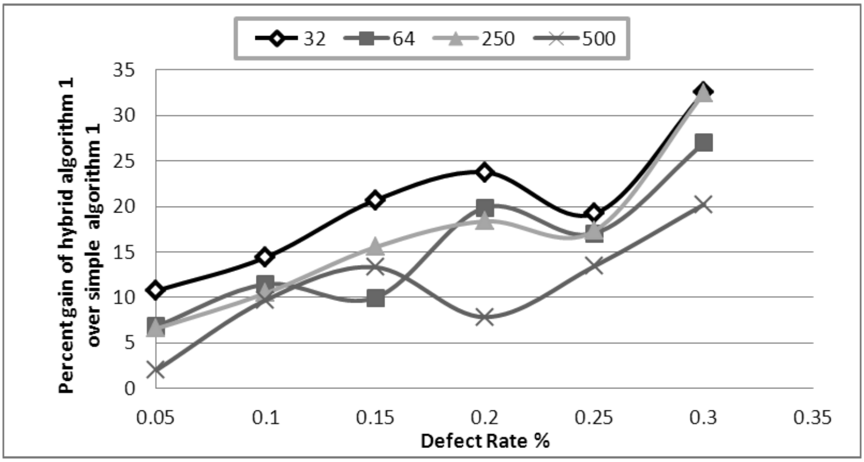

Fig. 3. Percentage improvement of results of hybrid algorithm 1 over simple algorithm 1 for various crossbar sizes

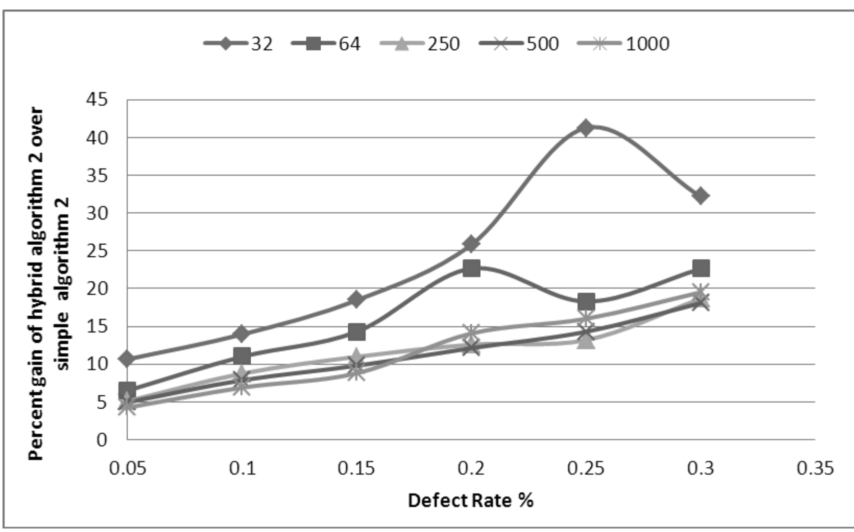

Fig. 4. Percentage improvement of results of hybrid algorithm 2 over simple algorithm 2 for various crossbar sizes

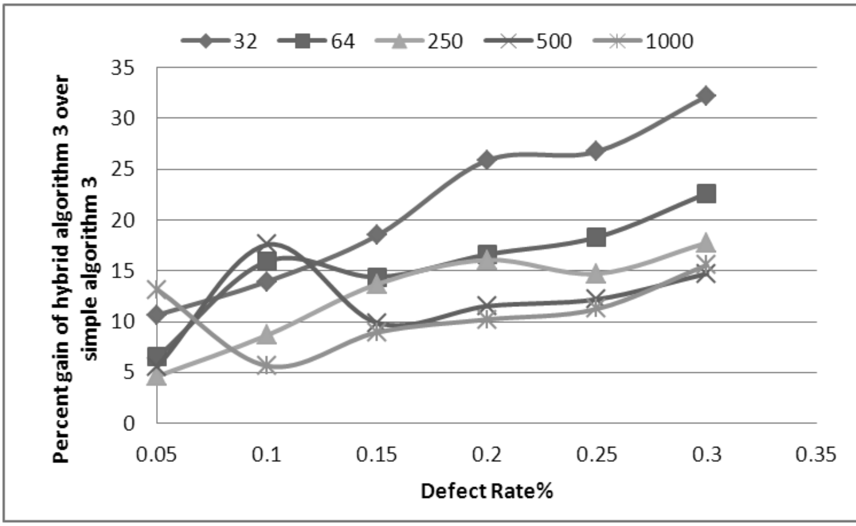

Fig. 5. Percentage improvement of results of hybrid algorithm 3 over simple algorithm 3 for various crossbar sizes

Figures 6-7 show the percentage improvement of hybrid algorithm 2 and hybrid algorithm 3 over simple algorithm 1, respectively. Figure 8 compares the effect of algorithm 3 after applying GA over algorithm 2. It is obvious that hybrid algorithm 3 is more efficient than simple algorithm 2. These results demonstrate that hybrid algorithm is very effective in improving the results as compared with the simple algorithms.

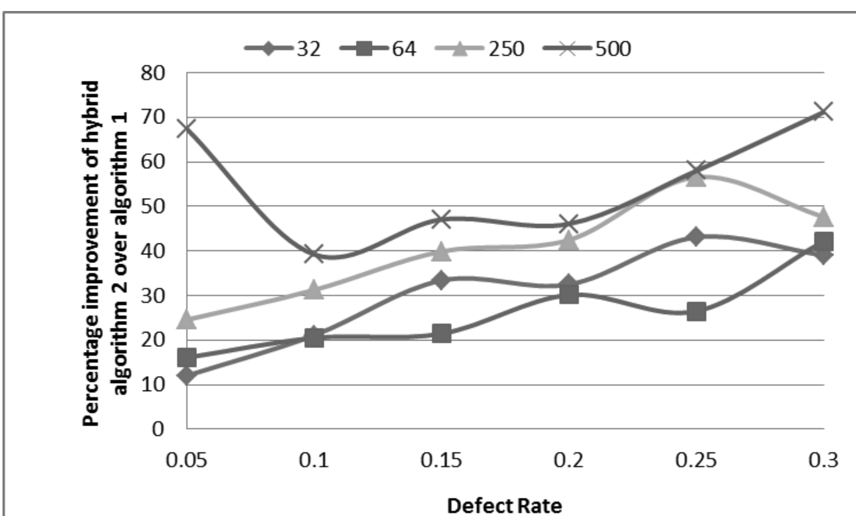

Fig. 6. Percentage improvement of results of hybrid algorithm 2 over simple algorithm 1 for various crossbar sizes

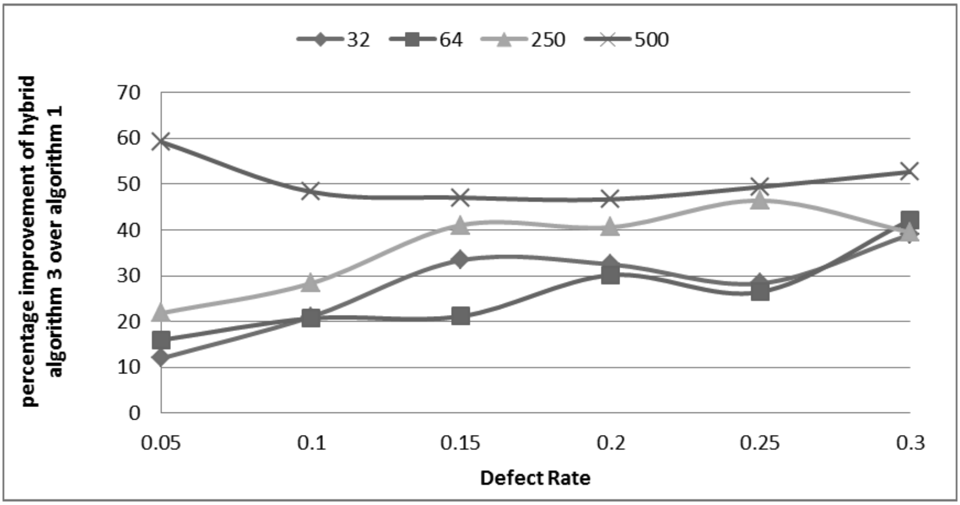

Fig. 7. Percentage increase of results of hybrid algorithm 3 over simple algorithm 1 for various crossbar sizes

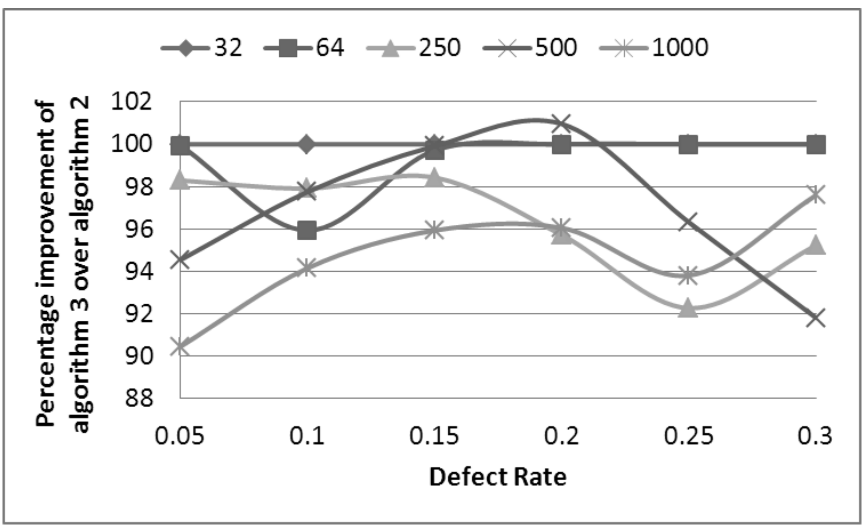

Fig. 8. Percentage improvement of hybrid algorithm 3 over algorithm 2 for various crossbar sizes

To measure the consistency of the resulting crossbars from the three algorithms before and after embedding with GA, the distributions of resulting crossbars (p = 15% for 250 x 250 crossbars) are shown in Figures 9-11. In these figures, n[k] is the number of crossbars of size equal to k. Distribution of the resulting crossbars for the three greedy algorithms after applying GA, were better than those before applying GA. The range of defect-free sub-crossbars spanning of the three greedy algorithms after applying GA are consistently higher than those spanned by those before applying GA. Regarding the CPU run time, the proposed algorithm spends more time as compared with the greedy algorithms. This is because of the repetitive application of heuristic for each chromosome of the population. However, the solution quality is substantially better.

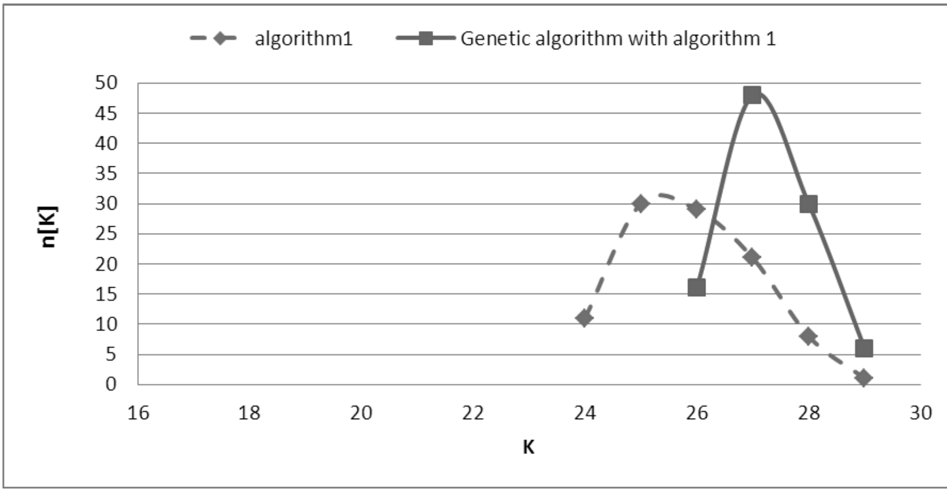

Fig. 9. Distribution of the resulting crossbars from simple algorithm 1 and the combination of GA and algorithm 1 for 250x250 crossbars with 15% defect rate

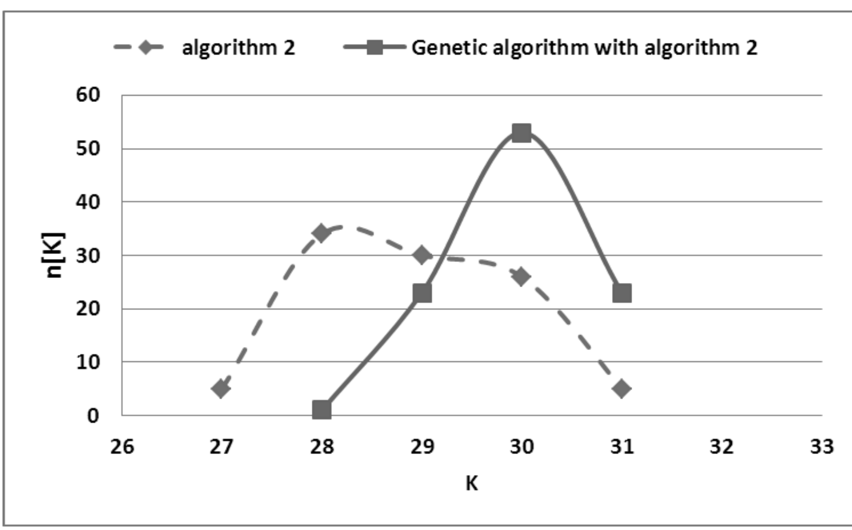

Fig. 10. Distribution of the resulting crossbars from simple algorithm 2 and the combination of GA and algorithm 2 for 250x250 crossbars with 15% defect rate

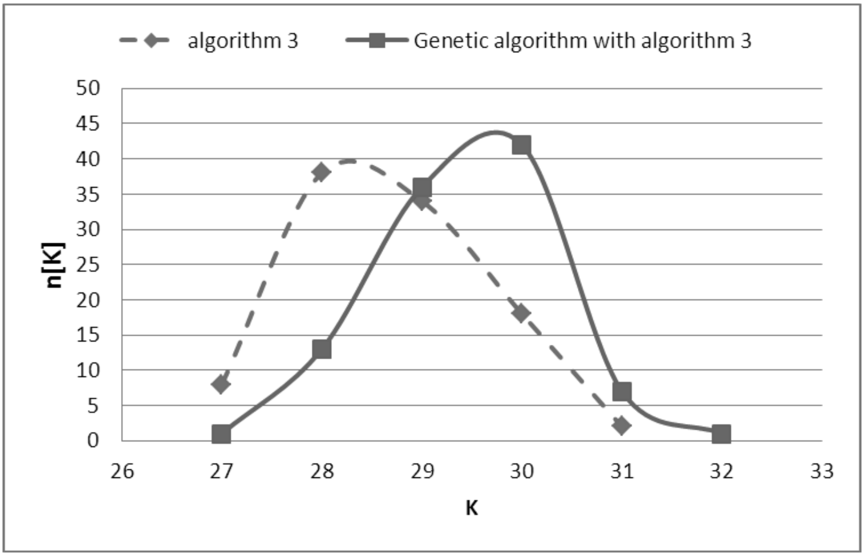

Fig. 11. Distribution of the resulting crossbars from simple algorithm 3 and the combination of GA and algorithm 3 for 250x250 crossbars with 15% defect rate

CONCLUSION

Defects are new challenges that arise when using nanotechnologies. The node selection order plays a key role in the final solution quality in the state-of-the-art heuristics for identifying the maximum size defect-free sub-crossbars in defective crossbars. The proposed hybrid method utilizes a new search space by integrating a known fast problem specific heuristic with the genetic algorithm. Since the node's degree plays a significant role in the selection mechanism of the greedy mapping heuristics, our proposed algorithm utilizes the node degrees and changes the selection order in the known mapping algorithms to find a better quality solution. The experimental results showed that the proposed hybrid algorithm was very effective in extracting a large size defect-free sub-crossbar in comparison with the other state-of-the-art algorithms. The run time of the proposed algorithm may be reduced by parallelizing it since the same operations are performed on different chromosomes to get different results. Moreover, any other low complexity greedy heuristic may be embedded with the proposed technique.

REFERENCES

Al-Yamani, A. A., Ramsundar, S., & Pradhan, D. K. 2007. A defect tolerance scheme for nanotechnology circuits. IEEE Transactions on Circuits and Systems-I **54(11)**: 2402-2409.

Bonam, R., Kim, Y. B., & Choi, M. 2007. Defect-tolerant gate macro mapping and placement in clock-free nanowire crossbar architecture. In Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems. pp. 161-169.

Dhodhi, M. K., Ahmad, I., Ahmad. I., & Yatama, A. 2002. An integrated technique for task matching and scheduling onto distributed heterogeneous computing systems. Journal of Parallel and Distributed Computing **62(9)**: 1338-1361.

Ghavami, B., Tajary, A., Raji, M., & Pedram, H. 2010. Defect and variation issues on design mapping of reconfigurable nanoscale crossbars. In Proceedings of the IEEE Annual Symposium on VLSI. pp. 173-178.

Gören, S., Ugurdag, H. F., & Palaz, O. 2011. Defect-aware nanocrossbar logic mapping through matrix canonization using two-dimensional radix sort. ACM Journal on Emerging Technologies in Computing Systems **7(3)**, article 12.

Langguth, J., Manne, F., & Sanders, P. 2010. Heuristic initialization for bipartite matching problems. ACM Journal of Experimental Algorithmics **15**, 1.3.

Lim, T. Y. 2014. Structured population genetic algorithms: a literature review. Artificial Intelligence Review **41**: 385-399.

Magun, J. 1998. Greedy matching algorithms, an experimental study. Journal of Experimental Algorithmics **3**, 6.

Paul, S., & Bhunia, S. 2012. A scalable memory-based reconfigurable computing framework for nanoscale crossbar. IEEE Transaction on Nanotechnology **11(3)**: 451-462.

Peeters, R. 2003. The maximum edge biclique problem is NP-complete. Journal of Discrete Applied Mathematics **131(3)**: 651-654.

Rao, W., Orailoglu, A., & Karri, R. 2006. Topology aware mapping of logic functions onto nanowire-based crossbar architectures. In Proceedings of the 43rd ACM/IEEE Design Automation Conference. pp. 723-726.

Shrestha, A. M., Takaoka, A., Tayu, S., & Ueno, S. 2011. On two problems of nano-PLA design. IEICE Transactions on Information and Systems **E94-D(1)**: 35-41.

Storer, R. H., Wu, D. S. & Vaccari, R. 1992. New search space for sequencing problems with application to job shop scheduling. Management Science **38(10)**: 1495-1509.

Su, Y., & Rao, W. 2009. Defect-tolerant logic mapping on nanoscale crossbar architectures and yield analysis. In Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems. pp. 322-330.

Su, Y., & Rao, W. 2014. An integrated framework toward defect-tolerant logic implementation onto nanocrossbars. IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems **33(1)**: 64-75.

Tahoori, M. B. 2005. A mapping algorithm for defect-tolerance of reconfigurable nano architectures. In Proceedings of the IEEE/ACM International Conference on Computer-Aided Design. pp. 668-672.

Tahoori, M. B. 2006. Application-independent defect tolerance of reconfigurable nanoarchitectures. ACM Journal on Emerging Technologies in Computing Systems **2(3)**: 197-218.

Yang, J. S., & Datta, R. 2011. Efficient function mapping in nanoscale crossbar architecture. In Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems. pp. 190-196.

Yaun, B., Chen, H., & Yao, X. 2014. A new evolutionary algorithm with structure mutation for the maximum balanced biclique problem. IEEE Transactions on Cybernetics, DOI:10.1109/TCYB.2012.2343966.

Yuan, B., & Li, B. 2011. Low time complexity defect-tolerance algorithm for nanoelectronic crossbar. In

Proceedings of the International Conference on Information Science and Technology. pp. 143-148.

Yuan, B., & Li, B. 2014. A fast extraction algorithm for defect-free subcrossbr in nanoelectronic crossbar. ACM Journal on Emerging Technologies in Computing Systems **10(3)**, article 25.

Zamani, M., Mirzael, H., & Tahoori, M. B. 2014. ILP formulations for variation/defect-tolerant logic mapping on crossbar nano-architectures. ACM Journal on Emerging Technologies in Computing Systems **9(3)**, article 21.

Zheng, Y., & Huang, C. 2009. Defect-aware logic mapping for nanowire-based programmable logic arrays via satisfiability. In Proceedings of the Design, Automation & Test in Europe Conference & Exhibition. pp. 1279-1283.

Zieglar, M., & Stan, M. R. 2003. CMOS/Nano co-design for crossbar-based molecular electronic systems. *IEEE Transactions on Nanotechnology*, **2(4)**: 217-230.

Submitted: 20-05-2014

Revised: 22-11-2014

Accepted: 03-12-2014