An Orthogonal Local Search Genetic Algorithm for the Design and Optimization of Power Electronic Circuits

Tao Huang, Jian Huang and Jun Zhang (corresponding author), *MIEEE*

***Abstract*—In this paper, an orthogonal local search genetic algorithm (OLSGA) is proposed for the design and optimization of power electronic circuits. The genetic algorithm is accelerated with a fast local search operator that automatically adjusts the search direction and the step size. An experimental design method called orthogonal design is used to determine the most promising direction of the potential region in the local search. In each generation, the step size is adaptively expanded or shrunk according to whether there is a newly improvement in the given local region. As a result, with proper direction and step size, the local search operator is able to stride forward and provide better exploitation ability to speed up the convergence rate of the genetic algorithm. The proposed method is applied to design and optimize a buck regulator. The results in comparison with other published results indicate that our proposed algorithm is effective and efficient.**

I. Introduction

THE continuous development of power electronic circuits asks for powerful design and optimization techniques to satisfy the increasingly rigorous requirements. During the last three decades, small-signal models have been widely used in the modeling and design of power converters. By applying the state-space averaging method [1], [2], current injected equivalent circuit approach [3], and sampled-data modeling technique [4], an averaged time-invariant model can be derived to study the dynamic stability of the circuit around the operating point. Although these methods are simple and elegant, the detailed information of the circuit waveform in one switching cycle is ignored. Moreover, they are often applicable for predetermined switching sequence of circuit topologies only, thus not valid for large-signal analysis. As a result, it could be very difficult for the circuit designers to predict precisely the circuit performance under large-signal disturbances.

Recent studies on optimization methods indicate great effectiveness of evolutionary algorithms for the global optimization problems, as they are less dependent on the initial starting point of the search and do not need derivatives in explicit analytical form. Among various approaches, genetic algorithms (GAs) [5]-[8] have gained continuous attentions. A GA starts with an initial population which consists of a number of random candidate solutions. The evolution of population is achieved by employing three basic operators: selection, crossover and mutation to generate offspring. Selection attempts to apply pressure upon the population in a manner similar to that of natural selection found in biological systems. Crossover allows individual to exchange information with each other imitating sexual reproduction used by natural organisms. Mutation is used to randomly change the value of single bit within individual strings that implement global exploration of the feasible region.

Recently, a GA based optimization scheme for switching voltage regulators is proposed in [9]. Although GAs have been proven to be a powerful search mechanism, the algorithms still suffer from premature stagnation and low convergence velocity. These drawbacks make the algorithms difficult to locate the global optimum when dealing with complex problems with many local optima. For the optimization of power electronic circuits, these problems become even more crucial because the computations of the fitness function are intensive. The direct application of the simple genetic algorithm to the optimization problem requires a long time to obtain an acceptable result and the accuracy may not be high enough. In addition, the component tolerances and their effects on the performance of the circuit are ignored in the optimization process in [9]. In the manufactured circuit, all components have tolerances associated with them. As a result, the performance of each manufactured circuit will usually differ from that of the simulated nominal one. Although the nominal values fully satisfy the specifications, sometimes a great many of productions may violate the requirements giving a low manufacturing yield. Therefore, it is necessary to develop a powerful method for the comprehensive optimization of power electronic circuits.

Many studies have been focus on improving the searching abilities of GAs. One common strategy is the hybridization of GAs with local search techniques. The hybrid algorithm benefits from the combination of the global exploration abilities of GAs and the exploitation advantages provided by local search techniques. A number of methods have been introduced in the published literature such as hill-climbing [10]-[12], landscape approximation [13], and gradient descent [14]. They have been successfully applied to solve various kinds of problems and shown that they could outperform traditional GAs. However, additional function evaluations are always necessary for local search techniques to exploit the vicinity. For real-world applications such as circuit optimization in which certain amount of simulations have to be performed, the evaluation of fitness function is usually the most expensive part. It is therefore desirable to find an effective and efficient way for the local search to locate the nearby optimum in fewer function evaluations. An orthogonal local search method is introduced in this paper by

This work was supported by NSF of China Project No.60573066; and the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, P.R. China.

Authors are with the SUN Yat-sen University, Guangzhou, P.R.China.(Jun Zhang is the corresponding author, email:junzhang@ieee.org)

978-1-4244-1823-7/08/$25.00©2008 IEEE

incorporating the local search with an experimental design method called orthogonal design.

TABLE I
THE ORTHOGONAL ARRAY $L_9(3^4)$

Factor / Combination	1	2	3	4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

TABLE II
THE ORTHOGONAL DESIGN OF THE CHEMICAL EXPERIMENT

Factor / Combination	Reaction pressure	Chemical concentration	Amount of catalyst
1	100kPa	50%	1g
2	100kPa	60%	2g
3	100kPa	70%	3g
4	120kPa	50%	2g
5	120kPa	60%	3g
6	120kPa	70%	1g
7	140kPa	50%	3g
8	140kPa	60%	1g
9	140kPa	70%	2g

The orthogonal design, which is one of the most popular design methods, has received widely application on science research, manufacturing industries, agricultural experiments and quality management. Recently, many researchers have successfully combined the orthogonal design method into the GA. Zhang and Leung [15] proposed an orthogonal genetic algorithm (OGA) for the multimedia multicast routing problems. Leung and Wang [16] designed an orthogonal genetic algorithm call orthogonal genetic algorithm with quantization (OGA/Q) for continuous variables optimization. Different from the previous works in which the orthogonal design method is implemented in the crossover operator, in our work the method is used in a local search procedure named orthogonal local search. The orthogonal local search proposed in this paper is comprised of two main strategies, namely the orthogonal exploitation and the adaptive step size. The orthogonal exploitation enables the procedure to search in the best direction and the adaptive step size strategy automatically adjusts the search range in different situations. Consequently, a fast convergence to the nearby optimum can be achieved. A novel genetic algorithm named orthogonal local search genetic algorithm (OLSGA) is proposed for the optimization of the power electronic circuits. The OLSGA is fast because of the orthogonal local search procedure and it is more practical because a tolerance analysis is incorporated in the fitness function. Experiments are conducted for the optimization of a buck regulator and the results indicate that the OLSGA is effective and efficient.

The rest of this paper is organized as follows. Section II briefly introduces the orthogonal design method taking an example of chemical experiment. Section III presents the orthogonal local search technique including the orthogonal exploitation and the adaptive step size strategies. In Section IV, the OLSGA is developed by incorporating the GA with the proposed orthogonal local search. Section V shows the fitness function with tolerance analysis. The results of the optimization experiments are given in Section VI and the conclusion of this paper is drawn in Section VII.

II. ORTHOGONAL DESIGN METHOD

An experimental design method called orthogonal design is imposed in the proposed algorithm to determine a proper search direction for the local search. In this part, we firstly give a brief introduction to this experimental design method and explain how it can reduce the efforts in finding the best direction.

Let's consider the following example. In order to maximum the yield of a chemical product, three relevant variables have to be fixed: a) reaction pressure, b) chemical concentration, c) amount of catalyst. These variables are known as factors in the experimental design. Suppose there are three possible options for each factor:

a) 100kPa, 120kPa, 140kPa
b) 50%, 60%, 70%
c) 1g, 2g, 3g

These different choices of each factor are called levels. To obtain the most profitable combination of levels, one simplest way is to do the full-scale experiment which means to evaluate every possible combination. For the chemical experiment, such method takes $3^3 = 27$ times of tests. A full-scale experiment is able to provide the best combination, however, at the cost of large amount of tests. For bigger number of factors and levels, for example 6 factors and 5 levels for each factor, the number of experiments to be tested becomes $5^4 = 15625$ which is absolutely unacceptable. As a result, the orthogonal design is introduced to find small but representative combinations by using a series of orthogonal arrays.

Let $L_M(N^K)$ denotes an orthogonal array for K factors and N levels, in which "L" stand for a Latin square and M is the total number of combinations. The orthogonal array $L_9(3^4)$ is illustrated in Table I. For the chemical experiment mentioned above, the final design shown in Table II is obtained by removing the last column of $L_9(3^4)$ because there are only 3 factors in the problem we consider. As shown in Fig. 1, the experimental design is illustrated with a cube that consist of 27 nodes for the totally 27 combinations. Each dimension denotes a factor with three nodes representing the corresponding levels. As it can be observed in the figure, the 9 nodes selected by the orthogonal array scatter uniformly in the space since:

1) There are equally three testing points in every plane of the cube.
2) There is equally one testing point in every row or column of each plane.

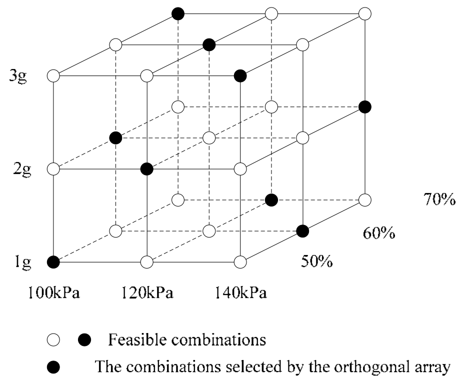


Fig. 1. Distribution of the combinations in a cube.

Hence, conclusion can be drawn that the orthogonal design is representative in reflecting the solution space, what is more important, with only a few selected combinations. In general, there are two main characteristics of the orthogonal array:

1) For any column of the array, every level occurs the same number of times.
2) For any two columns, every combination of two levels occurs the same number of times.

Based on these characteristics, more properties can be deduced:

3) With any two columns swapped, the resulting array is still an orthogonal one.
4) With any two rows swapped, the resulting array is still an orthogonal one.
5) With any two levels swapped, the resulting array is still an orthogonal one.
6) If any columns are deleted, the resulting array is still an orthogonal array.

The last four properties are called the basic fundamental transformations of orthogonal arrays. This kind of features makes the arrays more flexible as different orthogonal array can be used for the same problem. Usually, an experimental design that has a fix number of factors and levels can use orthogonal arrays with different number of combinations. Applying a larger array may achieve a more convincible result, but it suffers from a large quantity of experiments. On the other hand, using a smaller array is much faster in comparison, but the quality of the resulting solution may not be as good. Therefore, a tradeoff has to be made by choosing a suitable orthogonal array with proper scale of experiments so that a better combination can be found in a tolerable time.

III. Orthogonal Local Search

A neighborhood search procedure is able to deeper exploit the most promising region of the search space. Thus, the incorporation of GAs and local search techniques can greatly enhance the efficiency of the convergence to the global optimum. In order to fully take advantage of the local search, several questions have to be answered ahead of time, such as which individual should under go the local search; how to exploit the neighborhood effectively; how many efforts should be put in, etc. To the first question, it should be noted that the local search operator can be applied to every individual in the population. However, such kind of procedure requires large amount of function evaluations and the application of the local search to the ordinary individuals may turn out to be a waste of time. In the proposed algorithm, the global best solution is selected to exploit its neighborhood because it is more likely to be in the proximity of the global optimum and therefore be able to guide the search towards the target point.

In this paper, we mainly focus on two important issues – the search direction and the search range. When trying to seek for a better solution locally, we may attempt to slightly change a selected individual from its original location to see if there is any advancement. For the real-coded GAs, one common task is to decide whether a specific gene should be increased or decreased. Such issue becomes more crucial when multi decisions have to be made such as the K dimensions in the function optimization. This kind of decisions can be regard as the look for a direction in the solution space. Following a proper direction that provides maximum possibility of local improvement can greatly reduce the efforts in finding the global optimum. In contrast, searching in a bad direction may probably lead to a less potential region and turn out to be unnecessarily waste of function evaluations. It is therefore of vital importance to find out the best direction so as to enhance the ability of the local search operator. Another question to be solved is the possible amount of variation of each gene, that is, the range of neighborhood in the solution space. With a selected direction, an individual may take a step forward inside the neighborhood. An excessively large neighborhood may significantly reduce the possibility of finding better solutions as the individual may step over the global optimum. In contrast, searching in too small a local region can only result in tiny improvement and more steps have to be taken before reaching the global optimum which may slow down the pace. Therefore, it is integral to determine an appropriate step size to ensure both the efficiency and effectiveness of the local search operator. Two different strategies are designed to solve these problems, respectively.

A. Orthogonal Exploitation

The implementation of the local search to the global best individual can be regarded as an experiment. This angle of view inspires the application of the orthogonal design in the local search. For the orthogonal array $L_M(N^K)$, there are totally M combinations which stand for M search directions. As discussed before, these directions are more representative in reflecting other possible directions in the solution space. Let's consider an optimization problem with K factors and suppose that there are three levels for each factor: the value may 1) be increased, 2) be decreased, 3) remain unchanged. The global best individual denoted by $\mathbf{x}' = (x_1, x_2, x_3, \ldots, x_K)$ is selected to generate a number of M offspring $\mathbf{x}'_1, \mathbf{x}'_2, \mathbf{x}'_3, \ldots, \mathbf{x}'_M$ based on the corresponding orthogonal array $L_M(3^K)$ following the steps described below:

Step 1) For the offspring $\mathbf{x}'_j = (x_{1j}, x_{2j}, x_{3j}, \ldots, x_{Kj})$, each variable x_{ij} is generated by

$$x_{ij} = \begin{cases} x_i + r_i & \text{if } L_{ij} = 1 \\ x_i & \text{if } L_{ij} = 2 \\ x_i - r_i & \text{if } L_{ij} = 3 \end{cases} \quad (1)$$

where $L_{ij} \in [1,2,3]$ is the corresponding value of combination j with factor i in the orthogonal array $L_M(3^k)$ and r_i is the step size. As it can be concluded from (1), the offspring $\mathbf{x}'_j$ is generated according to the jth combination of the orthogonal array. In other words, each offspring represents a search direction.

Step 2)

$$x_{ij} = \begin{cases} x_{ui} & \text{if } x_{ij} > x_{ui} \\ x_{li} & \text{if } x_{ij} < x_{li} \end{cases} \quad (2)$$

where x_{ui} and x_{li} denote the upper and lower bound of x_i respectively. This step is to make sure the local search will not exceed the search domain of the optimization process.

Step 3) Evaluate each generated offspring by a predefined fitness function.

Step 4) Calculate the sum of fitness values for each level of each factor and select the levels with the largest sum to generate a predicted direction.

Step 5) Generate another offspring $\mathbf{y}' = (y_1, y_2, \ldots, y_K)$ according to the predicted direction by

$$y_i = \begin{cases} x_i + r_i & \text{if } B_i = 1 \\ x_i & \text{if } B_i = 2 \\ x_i - r_i & \text{if } B_i = 3 \end{cases} \quad (3)$$

where $B_i \in [1,2,3]$ represents the predicted level of factor i. Again, make sure the search domain is not violated.

Step 6) Evaluate the fitness value of the newly generated offspring.

Step 7) Pick out the best offspring and compare it to the global best individual. If the fitness value of the offspring is found to be higher, we may say that a promising direction is found and the global best individual will be substituted.

However, it must be noticed that this best offspring may not be the optimal individual in such search direction. A step forward with a proper step size is needed for further exploitation making full use of the information gathered by the orthogonal design. Moreover, if no better offspring can be found, the search range has to be adjusted so that a reasonable step size can be taken in the next generation.

B. Adaptive Step Size

In the proposed method, an appropriate step size r_i for each factor is very important for a successful search. As the best individual becomes closer to the global optimum during the optimization procedure, r_i should be shrunk to maximize the possibility of reaching the optimal solution or its neighborhood. On the other hand, if the step size is too small, the global optimum may not be contained in the local region and r_i should be expanded in such situation. To tackle this issue, we propose an adaptive method to adjust the step size taken in the orthogonal local search and the only steps are shown as follows:

Step 1) Initialize a steps counter N_{steps} to record the number of steps taken in a certain direction. Apply the orthogonal search to the global best individual. If a better solution is obtained, store the best direction and expand r_i for the next step

$$r_i = r_i / \lambda \quad (4)$$

where $\lambda \in (0,1)$ is a shrinking rate of the step size. Else if no better individual is found go to step 3).

Step 2) $N_{steps} = N_{steps} + 1$. If $N_{steps} > N_{\max}$, in which $N_{\max}$ is a user defined maximum number of steps, the local search procedure will be terminated. Otherwise, generate another offspring following the best direction based on the best so far individual. If the fitness value of this offspring is higher than the best so far individual, expand r_i according to (4) and repeat step 2). Or else, go to step 3).

Step 3) Shrink the step size by

$$r_i = r_i \times \lambda \quad (5)$$

and terminate the local search procedure.

From the steps above, we could see that if the orthogonal local search failed to provide any better offspring, r_i will be shrunk so as to narrow down the size of neighborhood. Otherwise, i.e. a promising direction is found, the procedure will further exploit the local region by a step forward for a better solution. The step size is expanded in such situation so as to go even deeper. The exploitation will be ended if there is no more improvement in such direction.

C. Implementation of the Orthogonal Local Search

The proposed orthogonal local search procedure is applied to the global best individual in every generation of the genetic algorithm. For the local search in each generation, the orthogonal array is reconstructed by random selection of the columns in the standard array for each factor. The new constructed array is still an orthogonal one according to the characteristics, but having different distribution in the solution space to make the search more general.

IV. Orthogonal Local Search Genetic Algorithms

The incorporation of GAs with local search techniques is able to compensate the deficiencies of GAs with the high exploitation ability of the local search. In this paper, an orthogonal local search genetic algorithm (OLSGA) is proposed by combining the GA with the proposed orthogonal local search. The rationale behind is to refine the potential individual to make it more sensitive to its neighborhood. The OLSGA here mainly follows the basic structure of the simple genetic algorithm.

The steps of the proposed OLSGA are described as follows with the aid of a flowchart shown in Fig. 2.

Step 1) Initialize the optimization parameters.

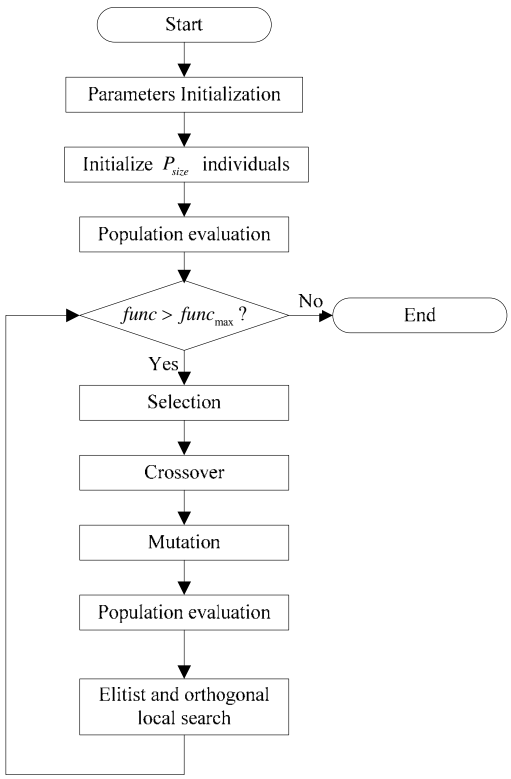


Fig. 2. Flowchart of the OLSGA.

Step 2) Randomly create a number of P_{size} individuals to buildup the first generation.

Step 3) Initialize a counter *func* which records the number of fitness function evaluations. Evaluate each individual in the population by a predefined fitness function.

Step 4) If $func > func_{\max}$, where $func_{\max}$ is the predefined maximum number of function evaluations, the optimization process will be terminated. Otherwise, go on to step 5).

Step 5) Select the individuals for the next generation using the roulette wheel selection method.

Step 6) Perform the crossover operator to the population with the crossover rate of p_x.

Step 7) Perform the mutation operator to the population with the mutation rate of p_m.

Step 8) Evaluate the new population.

Step 9) Reserve the global best individual and implement the orthogonal local search. Go to step 4).

V. Fitness Function

The proposed algorithm is used to design and optimize the same buck regulator as in [9]. The circuit schematic is shown in Fig. 3, in which the converter is decoupled into two parts,

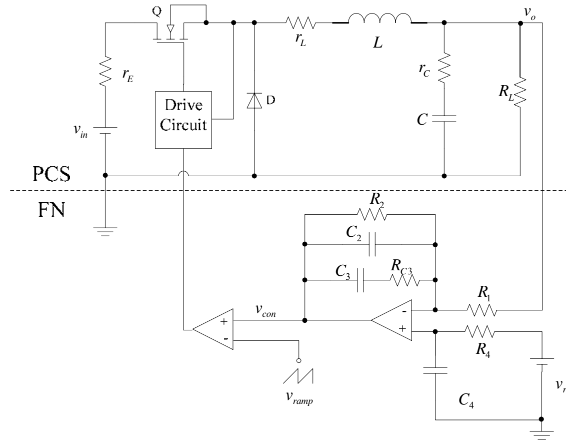


Fig. 3. Circuit schematic of a buck regulator.

namely the power conversion stage (PCS) and the feedback network (FN). These two parts are optimized separately in two different procedures. The PCS is optimized for the required static characteristics and the FN is optimized for both the static and dynamic behavior of the whole system. For the PCS, L and C are the design components with all the parasitic resistances being known a priori. All the components in the FN are the parameters to be optimized. In this paper, we are not going to apply the OLSGA to optimize the PCS part. Instead, the results C=1054μF, L=194μH in [9] are taken for the optimization of the FN. The fitness function of the FN which is taken from [9] is shown as follows

$$\Phi_F(CF) = \sum_{R_L = R_{L\_\min}, \delta R_L}^{R_{L\_\max}} \sum_{v_{in} = V_{in\_\min}, \delta v_{in}}^{V_{in\_\max}} [OF_5(R_L, v_{in}, CF) + OF_6(R_L, v_{in}, CF) + OF_7(R_L, v_{in}, CF)] + OF_8(R_L, v_{in}, CF) \tag{6}$$

where $CF = [R_{C3}, C_2, C_3, R_2, C_4, R_4, R_1]$ denotes the individual of the FN. $R_{L\_\min}$ and $R_{L\_\max}$, $V_{in\_\min}$ and $V_{in\_\max}$ are the minimum and maximum values of R_L and v_{in}, respectively. δR_L and δv_{in} are the steps in varying R_L and v_{in}. OF_5, OF_6, OF_7 and OF_8 represent the four object functions for the FN.

However, component tolerances are not considered in this fitness function. As a result, the resulting component values may not be the optimal ones for manufactured circuits. In this paper, we are not going to directly apply this fitness function. Instead, a more general one with tolerance analysis is adopted in which a sampling method is used to evaluate the influence of the component tolerances. The fitness function used in this paper is shown as follows

$$\Psi_F(CF) = [\Phi_F(CF) + \Phi_F(CF \times 1.05) + \Phi_F(CF \times 0.95) + \Phi_F(CF \times 1.03) + \Phi_F(CF \times 0.97)]/5 \tag{7}$$

TABLE III
OPTIMIZATION RESULTS

Algorithm	$func_{max}$	Mean	Best	Worst
OLSGA	3000	139.42	140.46	138.77
GA	15000	134.26	140.12	128.39

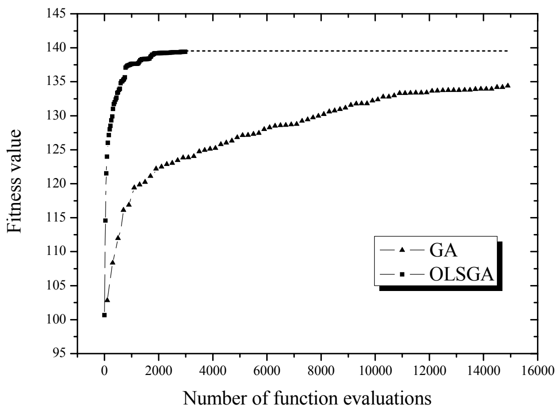
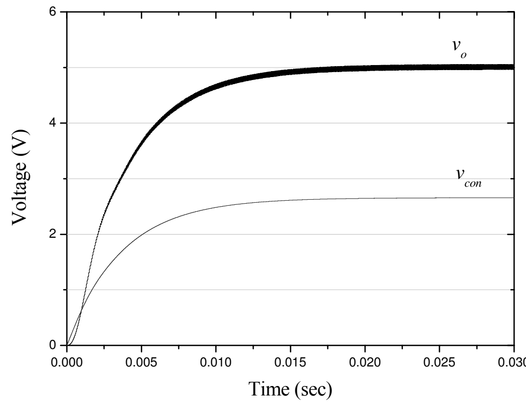


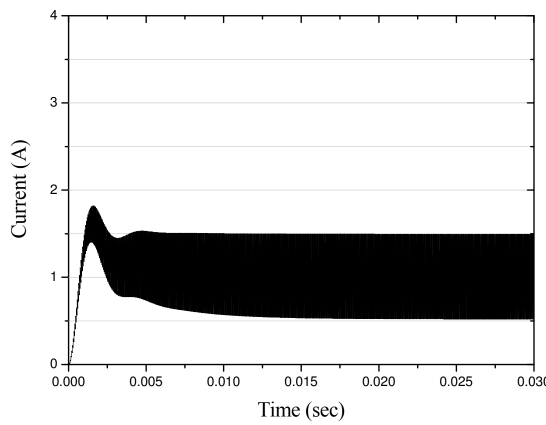
Fig. 4. Fitness value versus the number of function evaluations. (A dot line is drawn because the maximum number of function evaluations of the proposed OLSGA is 3000 only.)

TABLE IV
RESULTING COMPONENT VALUES OF THE BUCK REGULATOR

Component	Value
R_{C3}	44.298kΩ
C_2	16.409μF
C_3	0.669μF
R_2	2.897MΩ
C_4	4.024μF
R_4	1.845kΩ
R_1	449.939Ω



(a) v_o and v_{con}



(b) i_L

Fig. 5. Simulated startup transients when v_{in} is 20V and R_L is 5Ω.

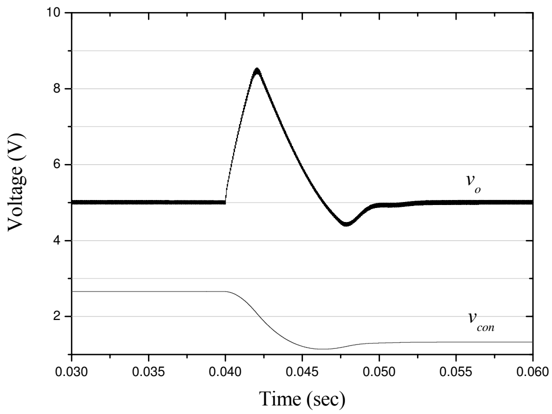
As we could see in the function, a small number of samples around the nominal values are evaluated and incorporated in the fitness function in order to give a general estimation of the circuit performance. It should be noticed that more samples with various distribution could be added in the fitness function to make it more precise. However, more computational time is required for evaluations of the additional samples. Therefore, a tradeoff has to be made considering both the effect and the time limit.

VI. EXPERIMENTS AND RESULTS

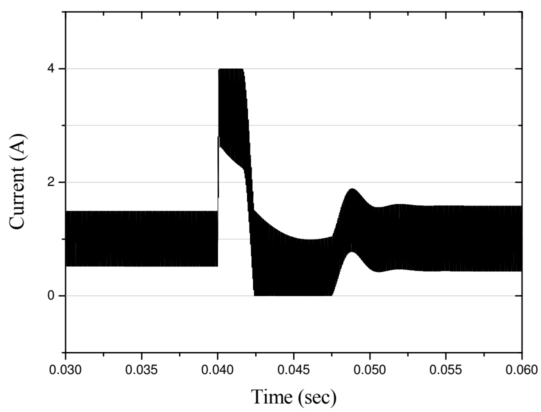
Experiments are carried out for the proposed OLSGA in comparison of the GA method used in [9]. In this part, we compare these two algorithms based on the fitness function evaluations. This is because the evaluation of fitness function is usually the most expensive part in the optimization of power electronic circuits. With the same number of function evaluations, the execution time of different algorithms are almost the same.

The parameters of the GA are set the same as in [9] where $P_{size} = 30$, $p_x = 0.85$, $p_m = 0.25$, except that the maximum number of fitness function evaluations $func_{\max}$ is 3000. The parameter settings of the orthogonal local search procedure are $N_{steps} = 2$ and $\lambda = 0.8$. To make fair comparison, each algorithm is executed for twenty independent trails.

The optimization results of both algorithms are tabulated in Table III. The 'Mean' stands for the average resulting fitness value of the twenty trials and the 'Best' and 'Worst' stand for the best and worst fitness value, respectively. As shown in the table, the OLSGA on average achieves better results than the GA method. Moreover, the proposed algorithm is more reliable as the worst fitness obtained is 138.77, which is much higher than the worst result of the GA. Fig. 4 shows the average fitness values versus the number of fitness function evaluations of both the results obtained by the GA and the

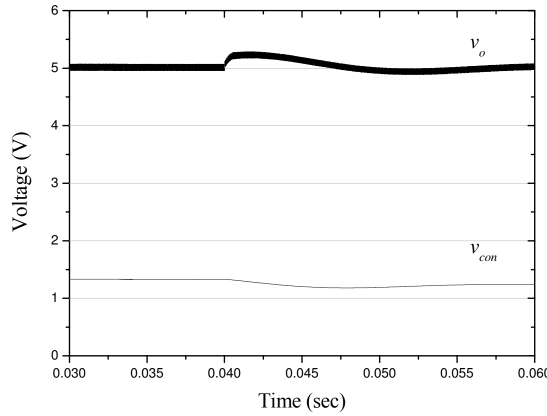


(a) v_o and v_{con}

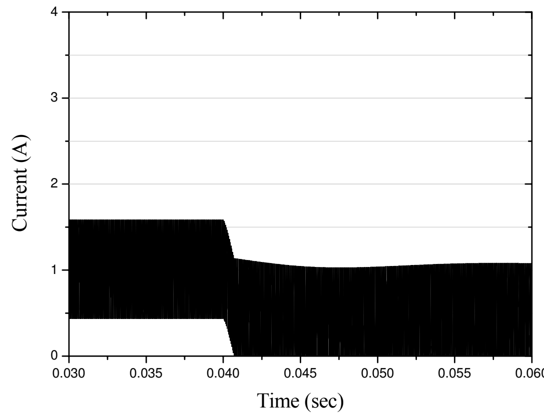


(b) i_L

Fig. 6. Simulated transient responses when v_{in} is changed form 20V to 40 V.



(a) v_o and v_{con}



(b) i_L

Fig. 7. Simulated transient responses when R_L is changed form 5Ω to 10Ω.

proposed OLSGA. As it can be observed in the figure, in the optimization process the OLSGA converges much faster than the GA. The optimal result is achieved within 3000 function evaluations in the proposed algorithm, whereas in the GA it requires as many as 15000 function evaluations and the global best fitness value is still not comparable with that of the OLSGA.

Circuit simulations are carried out for the resulting component values of the proposed OLSGA which is shown in Table IV. The simulated startup transients when the input source is 20V and the output load is 5Ω are shown in Fig. 5. As we can see in the figures, the settling time is less than 20ms and the steady state ripple voltage is less than 1%. No overshoot or undershoot can be observed in the output voltage. Large signal behavior tests are performed for the circuit and the input voltage is suddenly changed from 20V to 40V in the steady state. The transient responses are shown in Fig. 6. The output voltage can revert to the steady state in less than 20ms. Again, similar tests are carried out for output load disturbances when the input voltage is fixed at 40V. Fig. 7 shows the transients when R_L is changed from 5Ω to 10Ω. The steady state can be achieved when the circuit is working in discontinuous conduction mode (DCM). The results of the simulations show that both the static and dynamic responses are well within the required specifications. In comparison with the GA, the optimal circuit parameters can be obtained in a much shorter time in our program which confirms the virtues of the proposed method.

VII. Conclusion

A novel local search method is developed for the genetic algorithm to design and optimize the power electronic circuits in this paper. The advantages of the proposed algorithm lie in the detection of the best search direction and the adaptive step size strategy. The introduction of the orthogonal design method ensures the exploitation ability of the local search procedure by collecting representative combinations in the solution space. The most suitable step size is acquired during the adaptive expanding or shrinking of the search range. The design and optimization of a buck regulator is used to test the quality of the proposed method. The results are compared to the published results in the literature which verify both the effectiveness and the efficiency of the proposed OLSGA.

REFERENCES

[1] R. D. Middlebrook and S. ´ Cuk, "A general unified approach to modeling switched converter power stage," in *Conf. Rec. IEEE Power Electronics Specialists Conf.,* pp. 18–34, June 1976.

[2] F. C. Lee, R. P. Iwens, Y. Yu, and J. E. Triner, "Generalized computer-aided discrete time-domain modeling and analysis of dc-dc converters," *IEEE Trans. Ind. Elect. Contr. Instrum.,* vol. IECI-26, pp. 58–69, May 1979.

[3] P. R. K. Chetty, "Current injected equivalent circuit approach to modeling switching dc-dc converters," *IEEE Trans. Aerosp. Electron. Syst.,* vol. AES-17, pp. 802–808, Nov. 1981.

[4] G. C. Verghese, M. E. Elbuluk, and J. G. Kassakian, "A general approach to sampled-data modeling for power electronics circuits," *IEEE Trans. Power Electron.*, vol. PE-1, pp. 76–88, Apr. 1986.

[5] J. H. Holland, *Adaptation in Nature and Artificial Systems*, 2nded. Cambridge, MA: MIT Press, 1992.

[6] M. Srinivas, L. M. Patnaik, "Genetic Algorithms: A Survey," *IEEE Computer*, vol. 27 no.6, pp.17-26, June 1994.

[7] D. E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning," *Reading*, MA: Addison-Wesley, 1989.

[8] Z. Michalewicz, *Genetic Algorithm + Data Structures = Evolution Programs,* Springer-Verlag Berlin Heidelberg 1996.

[9] J. Zhang, H. S. H. Chung,W. L. Lo, S. Y. Hui, and A. K. M.Wu, "Implementation of a decoupled optimization technique for design of switching regulators using genetic algorithms," *IEEE Trans. Power Electron.*, vol. 16, pp. 752–763, Nov. 2001.

[10] D. Whitley, S. Gordon, and K. Mathias, "Larmarckian evolution, the Baldwin effect and function optimization," in *Parallel Problem Solving from Nature-PPSN III*, Berlin: Springer-Verlag, pp. 6–15, 1994.

[11] J. M. Renders and H. Bersini, "Hybridizing genetic algorithms with hill-climbing methods for global optimization: Two possible ways," *Proc. 1st IEEE Int. Conf. Evolutionary Computation*, pp. 312–317, 1994.

[12] J. Yen, J. C. Liao, B. Lee, and D. Randolph, "A hybrid approach to modeling metabolic systems using a genetic algorithm and simplex method," *IEEE Trans. Syst., Man, Cybern. B*, vol. 28, pp. 173–191, Apr. 1998.

[13] K.-H. Liang, X. Yao, and C. Newton, "Combining landscape approximation and local search in global optimization," *Proc. 1999 IEEE Int. Congr. Evolutionary Computation*, pp. 1514–1520, 1999.

[14] R. Salomon, "Evolutionary algorithms and gradient search: Similarity and differences," *IEEE Trans. Evol. Comput.*, vol. 2, pp. 45–55, July 1998.

[15] Q. Zhang and Y. W. Leung, "Orthogonal genetic algorithm for multimedia multicast routing," *IEEE Trans. Evol. Comput.*, vol. 3, no. 1, pp. 91–96, Apr. 1998.

[16] Y. W. Leung and Y. P. Wang, "An orthogonal genetic algorithm with quantization for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 5, no. 1, pp. 41–53, Feb. 2001.