Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 6, Issue, 12, pp.10490-10493, December, 2014

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

ISSN: 0975-833X

RESEARCH ARTICLE

A MODIFIED SIMPLE GENETIC ALGORITHM FOR EVOLUTIONARY IMAGE FILTERS

[1*]Nirmala, P. and [2]Dr. Rani Hemamalini, R.

[1]Research Scholar, St.Peter's University
[2]Professor, St.Peter's College of Engineering and Technology, Anna University, India

ARTICLE INFO

Article History:
Received 24th September, 2014
Received in revised form
26th October, 2014
Accepted 09th November, 2014
Published online 27th December, 2014

Key words:
Evolutionary algorithm,
Genetic algorithm,
Impulse noise filtering,
Generation run.

ABSTRACT

Evolutionary algorithms are widely used optimisation technique for an efficient combinatorial design in the field of evolvable hardware. This paper deals with the design of non-linear image filter using genetic Algorithm. First, a modified simple genetic algorithm is proposed which is based on the concept of generation runs and shuffling operation. This generation run polls for the best fit offspring to pass on to the next generation along with retaining the elite population. This enables the Genetic algorithm to search the sub spaces of the large solution space. The shuffling operation at the end of each generation will avoid the problem of positional locality reference which causes the local minimum problem. A sample test function is employed to guarantee the convergence of sGA to an optimum solution. Second, the proposed modified sGA is capable of designing an efficient image filter for the impulse noise filtering. The suitable impulse noise modelling that reduces the computational complexity is also employed in the filter design.

INTRODUCTION

Evolvable hardware deals with the techniques of reconfiguring hardware modules based on evolutionary techniques. These techniques mimic the natural selection processes that are robust and powerful adaptive search mechanisms (Pradeep *et al.*, 2010) that are useful for system design. The techniques are well suited for the many non linear applications like image filter design in image processing, configuration of reconfigurable devices. This paper deals with design of evolutionary image filter for impulse noise filtering. Such filter design is highly adaptive to real time noisy environments. During image transmission, images are being increasingly corrupted by various types of noises resulting in significant loss of information. In order for the filtering operation not to disturb the noisy pixels, noise estimation technique is employed. The evolutionary technique used for the image filter design is Genetic Algorithm. (GA). Genetic Algorithm is a stochastic method which traverses the entire fitness landscape to reach the minimum or maximum point using suitable genetic operators. Genetic algorithm starts with a random set of population from a large solution space. A suitable fitness criterion is employed to check the fitness of these solutions. Based on these fitness score, the population is evolved through various generations after applying suitable genetic operators. The GA flow is shown in Fig. 1. This continues till the stopping condition is met or fitness value remain constant over generation.

Corresponding author: Nirmala, P.
Research Scholar, St.Peter's University

The best fitness chromosome represents the optimum solution to the given problem. After each generation the updates are made towards the best fitness point. The main aim to develop modified simple genetic algorithm (GA) with suitable constraints. The proposed sGA and its efficiency in the design of image filter using evolvable hardware technique (Xin Yao, 1999). The GA control parameters like population size, mutation rates and crossover probabilities plays a vital role in taking the GA search to reach an optimum solution. These determine the path taken by the GA search in reaching the better fit point. For the reference, μ represents parent with δ the off spring and j generation. The updating equation is given as;

$$pop_{j+1}(\mu) = pop_j(\mu' + \delta)$$

The paper describes the following sections as follows. In section II, a modified sGA is developed with generation runs technique and shuffling operation alongside a test function is explained for its convergence measure. Section III describes the impulse noise modeling of salt and pepper noise for noise pixel estimation. In section IV Evolutionary design of image filter and its sGA constraints with the results and discussion in section V and finally the conclusion with future scope for research is discussed in section VI.

A Modified Simple Genetic Algorithm

In steady state GA, only a part of the population is updated with elite offspring instead of whole population. GA iteratively searches the solution space using suitable genetic operators.

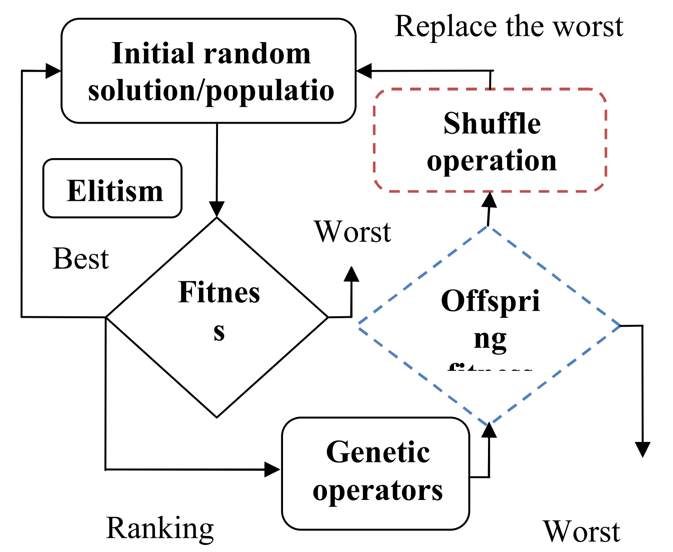

Fig. 1. GA flow cycle

The main motive of these operators is to make GA to visit most possible points in the space without getting locked at local minimum point. This modified sGA invokes generation run technique which selects the best offspring with best fitness score to pass it to successive generation. The GA has following stage, initial random population, fitness measure, mutation crossover, selection and replacement (Pei-Yin Chen *et al.*, 2008).

Chromosome representation

For any i represent any point with $pop(i)\epsilon$ Mapped to Θ *(solution space)* and $pop(i) \subset \mathbb{R}$; real coded chromosome. The initial population is randomly generated which represents the part of solution space. The chromosome encoding fixes the solution space for a given problem. For example, if a chromosome has length ℓ= 40, then the search space corresponds to 10^{40}. This initial population follows uniform probability distribution which is given by

$$for\ j = 1\ to\ n\ \ \{\mathcal{U}(pop_j) = 1/n\}.$$

Each chromosome comprises of genes and in turns has alleles which are a genotypic representation of a phenotypic character.

Generation and Generation runs

Initially, $\mathcal{N}$ denotes the generations limit with a represents the runs for each generation. This run is to poll for the best offspring generated with the crossover probability $\mathcal{P}_c = 0.5$ and mutation probability $\mathcal{P}_m = 0.2$. These run helps to explore the sub space of the solution space Θ. The possible solution space possible within finite time is $\mathcal{N}^a$ with this generation runs and generation count. The crossover probabilities about a point eth will generate two offspring as following

$$off1 = pop1(1{:}e) + pop2(e + 1{:}\ell);$$
$$off1 = pop1(1{:}e) + pop2(e + 1{:}\ell);$$

The mutation will generate the population with a part of the chromosome being replaced with a new gene.

$$off = pop(1{:}\mathcal{Z}) + rand(\mathcal{Z}{:}\ell);$$

The generation run makes sure that the offspring of better fitness is passed to the next generation. Fig. 2 shows the how generation run could make single parent to visit many possible solution point of space Θ through its generation run best offspring. The replacement of these offspring to the population is replacing the worst fit$pop(i)$.

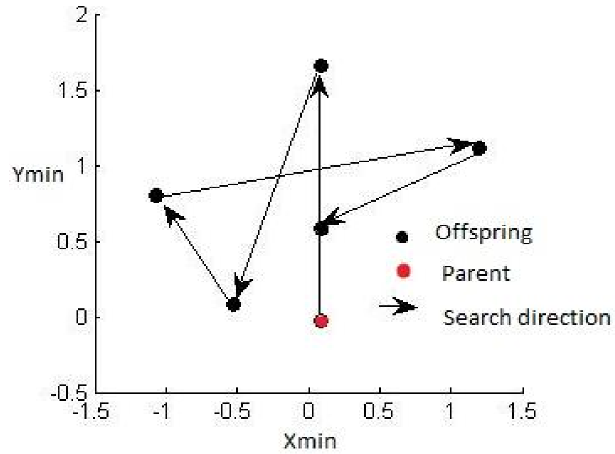

Fig. 2. Generation run makes the parent to visit diverse solution point through its off springs

Elitism and Rank selection

The best four elite solutions are retained with generation run best offspring replacing the worst fitness chromosome in the population based on their ranking. The elite offspring resulted after generation run will be added to the population. Ranking splits the optimum solution search space and the other one. The rank based selection makes the updated population

$$new_{pop(j)} = rank\ (fitness\{pop_{[j]}\});$$

These elite chromosomes with the elite offspring will form a part of population which serves as the current population for next generation.

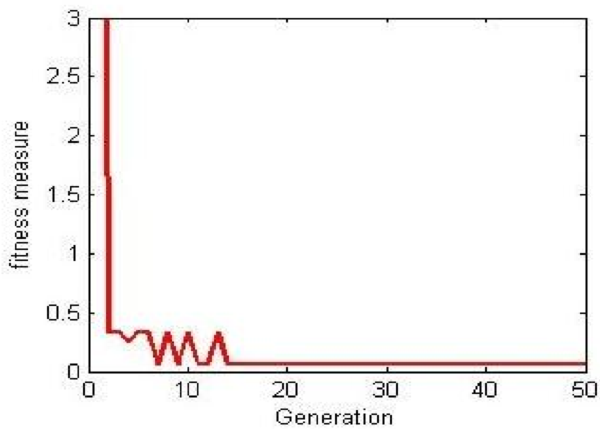

Fig. 3. Convergence of fitness measure vs. Generation

Objective function

Rosenbrock functionis a nonconvex function that has parabolic flat valley whose convergence to global minimum point is trivial *is* considered for the test which is described as

$$y(n) = 100(x^2 - y)^2 + (1 - y)^2$$

Constraints: $-2 < (x, y) < 2$

Theoretical minimum point located at (1, 1) whereas with sGA yields the optimum Point locating at (.9782, 0.9669) function value reaching its lowest minimum to 0.3276

Impulse Noise Modeling

Images are often being corrupted by Impulse noise represents the most common type of generic model in image noise analysis. This may be due to error in the channels or sensors. There are two variant of impulse noise (Vasicek *et al*., 2010), one is random noise whose value can vary anywhere in the dynamic range while the other on is salt and pepper noise which corresponds to the [Min, Max] of the grey scale intensity. The impulse noise corrupted region will vary significantly from its neighbor pixels. The sum of absolute difference between the centre pixel and the surrounding will be compared with the median value of the 3 × 3 window sized pixel values. If it is greater than threshold then the centre pixel is noisy else it is noise free. The image *(i, j)* varies with the median as threshold value.Absolute difference is

$$\sum\sum_{j,i}^{m,n} |difference| < median \begin{cases} noise\ less \\ noisy \end{cases}$$

There are various other robust statistics (Vasicek *et al*., 2010) which can accurate model the impulse noise. For evolutionary replacement of the noisy pixels, this simple model is simulated.

Image Filter Evolution

The functional level evolution (Sekanina, 2006) of image filter using sGA involves the use $\mathcal{U}\, rows(4) \times \mathcal{V}\, columns(5)$ reconfigurable array where the hardware module having the functions as shown in Table 1.

Table 1. Functional block of reconfigurable array

FUNCTIONS		
255-x	(x∩y)	Min(x,y)
(X+y)/2	(X+y+1)/2	Max(x,y
~x	(X+y)/2 + 1	~(x∩y)
~(x∪y)	x/2	x

The chromosome comprises of 20 pair of genes with one allele for the output making the value $\ell = 41$. The constraint considered is the input for a functional block should come from the previous column only with no crossing over. The hardware functionality is maintained constant while the interconnection pattern is evolved using Genetic Algorithm. The real coded chromosome is decoded to implement the effectiveness of image filter design. The initial population size considered is 30 and the generation run of 50 with both mutation and crossover operations taking place with $\mathcal{P}_c = 0.5$and $\mathcal{P}_m = 0.2$. For each generation, the best chromosome and its respective best fitness measure is recorded along with the offspring evolution. The other genetic algorithm scans the off springs after being updated in the original population. A 3×3 window is chosen and its neighborhood pixel is passed as inputs to the reconfigurable circuit and the output pixel is available on the 8bit output of the circuit. This output replaces the centre pixel. The entire array has the dimension of 4 (rows) * 5 (column) with programmable nodes at the nodes. The programmable nodes (Sekanina, 2004) operate on two inputs and produce one output which operates over 8 bits. The interconnection of these nodes is evolved using Genetic Algorithm to evolve suitable image filter for a given type of noise. The functionality of these nodes is chosen from this table and fixed while only interconnection are evolved. The fitness function chosen for evaluating the fitness of the population and offspring is absolute per pixel. (ADPP)

$$ADPP = \sum_{I=1}^{M}\sum_{J=1}^{N} |ev(i,j) - or(i,j)|$$

Where $ev(i,j)$ denotes the evolved image and $or\,(i,j)$ represents the original image, M and N are the size of the respective images.

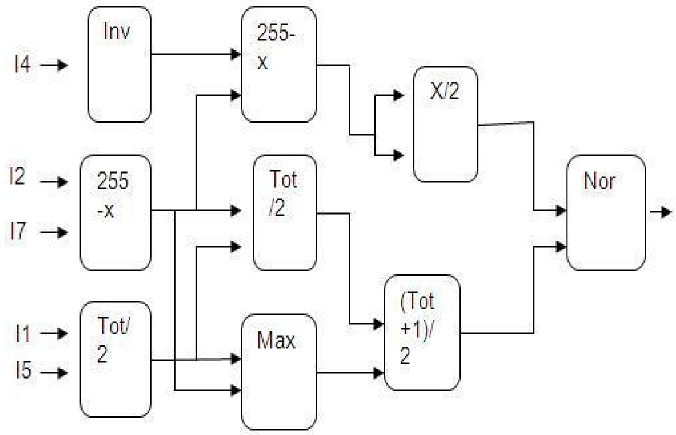

Fig. 4. The evolved image filter design

The fitness value of a candidate chromosome is obtained as follows:

- The circuit simulator is configured using a candidate chromosome.
- The absolute differences between pixels of the images are added.

RESULTS AND DISCUSSION

The modified sGA converge the Rosenbrock test function fitness measure to the optimum minimum solution which is very nearer to the actual value whose convergence property is shown in fig 3.The Evolutionary image filter schemepresented in this paper for the removal of salt-and-pepper noise using modified sGA which is more efficient in terms of computational complexity.

Fig. 5. Lena image noisy and the filtered one using the evolved filter

The optimum results achieved within 30^{th} generation with generation run of 50 each to generate better off springs. The resulted image filter structure is shown in Fig. 4. The test image to the evolved filter circuit and its filtered version is shown in Fig. 5. This shows that the evolved filter structure can efficiently perform the impulse noise removal along withthe impulse noise modeling.

Conclusion

This scheme is based on the noise estimation since it groups image pixels into different sets and deals with noise pixel by pixel instead of combining them as a whole. Thus modified sGA is effective in removing impulse noise which can be further improved by introducing various other statistics for the estimation problem. Additionally, the noise cancellation is a function level evolution based dynamic process in which no explicit target filter exists.

REFERENCES

Pradeep R. Fernando, Srinivas Katkoori, Didier Keymeulen, Ricardo Zebulon, and Adrian Stoica, 2010. "Customizable FPGA IP Core Implementation of a General-Purpose Genetic Algorithm Engine" IEEE *Transactions on Evolutionary Computation,* vol. 14, no. 1, February 2010

Vasicek, Z., L. Sekanina, and M. Bidlo. 2010. A method for design of impulse bursts noise filters optimized for FPGA implementations. In *DATE 2010: Design, Automation and Test in Europe*, pages 1731–1736. European Design and Automation Association, 2010

Sekanina, L. 2006. "Evolutionary Design of Digital Circuits: Where Are Current Limits?" *in Proc. the First NASA/ESA Conference on Adaptive Hardware and Systems 2006 (AHS 2006)*, pp.171-178, June 2006.

Sekanina, L. 2004. Evolvable components: From Theory to Hardware Implementations, *Springer,* 2004.

Pei-Yin Chen, Ren-Der Chen, Yu-Pin Chang, Leang-San Shieh, E, and Heidar A. Malki, 2008. Hardware Implementation for a Genetic Algorithm IEEE *Transactions on Instrumentation and Measurement,* vol. 57, no. 4, April 2008

Xin Yao, Tetsuya Higuchi, 1999. "Promises and Challenges of Evolvable Hardware" *IEEE Transactions on systems, man, and Cybernetics*—part c: applications and reviews, vol. 29, no. 1, February 1999

****