International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-3, Issue-3, March 2015

Novel Evolutionary Algorithm for ICA Processor for FPGA Implementation

Miss. G.Lakshmi Priya, Mr.A.raghuram

Abstract— **Evolutionary programming (EP) has been applied to many numerical and combinatorial optimization problems in recent years. Independent component analysis (ICA) is a statistical signal processing technique for separation of mixed signals, voices and images. The need for evolutionary algorithm for ICA lies in the fact that it needs contrast function optimization which enables the estimation of the independent components. Independent component analysis (ICA) decomposes observed mixed random vectors into statistically independent variables. It aims at finding the underlying independent components in the mixture by searching a linear or nonlinear transformation. It is also more efficient when the cost function, which measures the independence of the components, is optimized. ICA algorithm for contrast function optimization is developed in VHDL .The use of low complexity evolutionary computation with additional operations of mutation and crossover resolves the permutation ambiguity to a large extent. This also ensures the convergence of the algorithm to a global optimum and VLSI implementation results in reduced complexity of algorithms. IEEE single-precision representation, which fits in thirty-two bits, is used for all the manipulations for covering large range of real values.**

Index Terms— **ICA, Evolutionary optimization algorithm, FPGA , Statistical signal processing, VLSI**

I. INTRODUCTION

In the past few decades, there has been widespread interaction between researchers seeking various evolutionary computation methods to seek best solutions to a given function . Optimization algorithms have constituted most significant subjects in mathematics and industry to conceive more accurate and expeditious solutions. For all the traditional algorithms, optimization continues to pose a challenge in most real world cases because of large and complex solution space . There are still large-scale optimization problems that necessitate speedy resolution in a time span between ten milliseconds and a few minutes. Indeed, speed and precision are the main goals considered to be at variance. Optimization accuracy can be enhanced only if there is more time available. These obstacles, which are correlated with each other and with the utilization of mathematical operation, have paved the way for the Evolutionary Algorithm, first introduced population and selection of the next generation from the mutated and the current solutions.These two steps are coming under population-based version of the classical generate-and-test method. Mutation is a process of generating offspring and selection is used to test which of the newly generated solutions should go to the next generation. One disadvantage of evolutionary computation in solving some of the optimization problems is its slow convergence to a good near-optimum. A new mutation operator based on Cauchy random numbers is proposed and tested in . The new optimization with Cauchy mutation significantly outperforms the classical Evolutionary programming (CEP), which uses Gaussian mutation. This method provides few local minima, being comparable to CEP that has many local minima .The Evolutionary Algorithm was developed by mimicking or simulating processes found in nature and mainly includes Genetic Algorithms, Memetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, and Shuffled Frog Leaping Algorithm (SFLA).by Holland.Optimization by computation methods involves two major steps: Mutation of the solutions in the current

ICA (ICAPs) are essential for real-time processing of real-world digitized data, performing the high-speed numeric calculations necessary to enable a broad range of applications . Signal processing can be done in two categories of numbers. One is by using fixed point representation and another one is by using floating point representation. Since Signal processing techniques necessitate a large dynamic range of numbers, Fixed-point representations are unsatisfactory for most of the signal processing applications. The use of floating point helps to alleviate problems often seen in fixed point formats. There are two variants of floating point representation of a real number Based on the storage area available. IEEE single-precision representation that uses 32 bits and IEEE double-precision representation that uses 64 bits are the two variants . IEEE single precision format, that uses 32 bits, is used for ICA in this work.

Independent component analysis (ICA) is a statistical technique that plays an important role in a variety of signal and image processing applications such as blind source separation, recognition , and hyper spectral image (HSI) Analysis, blind de-convolution, and feature extraction[1] .A simple assumption on ICA is that, the observed signals are generally the linear combinations of the source signals. An example For ICA problem is cocktail party problem in which the acoustic signal captured from any microphone is a mixture of individual speakers speaking at the same time. Although powerful, ICA is very time consuming for implementation due to its computation complexity and the slow convergence rate, especially for high-volume or dimensional data set. All the existing ICA methods do not find a global optimum solution once the algorithm reaches a local optimum. It gets stuck in the valley of the contrast function and is unable to jump the surrounding hills .So a novel ICA algorithm using optimization technique is proposed to avoid getting trapped in local minima. Very large scale integration (VLSI) solutions with optimal parallelism

Manuscript received March 24, 2015.
Miss.G.Lakshmi Priya, Department Of Electronics and Communication Engineering Indira Institute of Technology&Sciences, Markapur.

Mr.A.raghuram, Department Of Electronics and Communication Engineering Indira Institute of Technology&Sciences, Markapur.

provide potentially faster and even real-time implementations for ICA algorithms [3]. During the last decade, advances in very large scale integrated (VLSI) circuit technologies have allowed designers to implement some ICA algorithms on fully analog CMOS circuits, analog–digital (AD) mixed-signal ICs, digital application-specific ICs (ASICs), and general field programmable gate arrays (FPGAs).The field programmable gate arrays (FPGAs) implementation of ICA VLSI processor with optimization technique is proposed that provides a potentially faster and real-time alternative. Contrast function plays a vital role in finding de-mixing matrix. So optimization technique is proposed for ICA for contrast function optimization [9,10].

II. SHUFFLED FROG LEAP ALGORITHM

The SFLA was recently devised as a novel meta-heuristic algorithm by Muzaffar Eusuff and Kevin Lansey. This algorithm is based on observing, imitating, and modeling the behavior of frogs searching for food placed in a pond. SFLA has been tested on a large number of combinatorial problems and found to be efficient in finding global solutions [14]. Furthermore, the SFLA compares favorably with the Genetic Algorithm, the Ant Colony Optimization, and the Particle Swarm Optimization in terms of time processing . The SFLA is a population-based cooperative search and consists of a frog leaping rule for local search and a memetic shuffling rule for global information exchange. In the SFLA, first an initial population of *F* frogs is created randomly. Next the population of *F* frogs is sorted in order of increasing performance level and separated into *m* memeplexes each holding *n* frogs in such a way that the first frog goes to the first memeplex, the second frog goes to the second memeplex, the *m*th frog goes to the *m*th memeplex, and the *(m+1)*th frog goes back to the first. The next step is the evaluation of each memeplex. In this step, each frog in the memeplex leaps toward the optimum location by learning from the best frog, so that the new position of the worst frog in the memeplex is calculated according to (1).

$$x\_worst^{k+1}=x\_worst^{k}+r^{k}(x\_sbest-x\_worst^{k})$$

where x_worst^{k} is the position of the worst frog in the memeplex, x_worst^{k+1} is the position of the best frog in the memeplex, r is a random number between 0 and 1, and k is the iteration number of the memeplex. If this evolution produces a better frog (solution), it replaces the older frog. Otherwise, the calculation of the new position can be expressed by (2):

$$x\_worst^{k}+r^{k}(x\_sbest-x\_worst^{k})$$

If non-improvement occurs in this case, a random frog is generated to replace the old frog .

A. Pseudocode of shuffled frog leap algorithm :

Begin;
Generate random population of P solutions
For each individual i
P: calculate fitness (i);
Sort the population P in descending order of their fitness;
Divide P into m memeplexes;
For i=1 to number of generations
    For each memeplex;
    Determine the best and worst frogs;
    Improve the worst frog position using Eqs. (1), (2)
End;
Combine the evolved memeplexes;
Sort the population P in descending order of the Check if termination=true;
End;
End;ir fitness;

III. INDEPENDENT COMPONENT ANALYSIS

Independent Component Analysis (ICA) is one ofthe most commonly used algorithms in blind source separation. It is the problem of finding unknown, unobserved or hidden structure in high dimensional data. Independent component analysis (ICA) is a technique of data transformation that finds independent sources in recorded mixtures of sources. It does not require any information on incoming signals. Since it utilizes only the statistical independence of the incoming signals, this separation problem is known as blind signal Separation. Such techniques have been applied in many fields, such as biology, biomedical signal processing, digital communication, and speech processing. H. Du, H. Qi and X. Wang compared different VLSI architectures given by various authors .

All the existing ICA methods do not find a global optimum since it may get stuck with local optimum. In addition to the problem of getting trapped in a local optimum, these algorithms have the ambiguities like scaling and permutation. The performance of all available algorithms depends on contrast functions that is the function of statisticalindependence .There are different contrast functions used forICA. The most popular contrast function used in ICA is kurtosis which measures the independency. Amit Acharyya and Koushik designed hardware Efficient Fixed-Point VLSI Architecture for 2D Kurtotic FastICA[2].

Evolutionary computation techniques are very popular population search based optimization methods . Genetic Algorithms, Swarm intelligence, Bacterial Foraging Optimization and Shuffled Frog Leap Optimization Algorithm are the most widely used evolutionary computation based optimization techniques . Instead of updating the matrix by a fixed formula as in FAST ICA, these evolutionary mechanisms can be used to search for the optimal separating matrix that minimizes the dependence. The block diagram of ICA is shown in Fig.1.The relationship between source signals S and observed mixtures X is given in(3) in matrix notation.

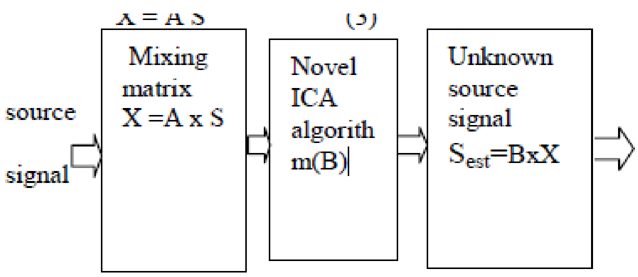

Fig 1: Proposed ICA Block diagram

A is a full rank matrix which is called mixing matrix. Under some assumptions, ICA solves the BSS problem by finding inverse linear transformation such that, it maximizes the statistical independence between the observed mixtures. For doing this, ICA finds de-mixing matrix B so that

S_est=BX=S-------------------- →(4)

Source matrix according to (4)

A. *ICA Preprocessing:*

It is highly recommended to perform preprocessing before applying the ICA algorithm in order to simplify the estimation process. The preprocessing of mixed signal involves finding the mixing matrix P.

a) Centering:
The first step in preprocessing is called Centering. It consists of subtracting mean from each observed mixtures as shown in Fig.2

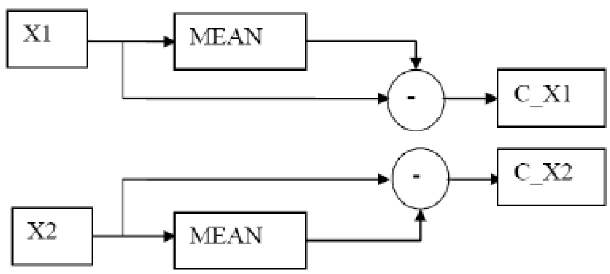

Fig.2 Implementation of Centering

b)Whitening:

The second step is called Whitening and it consists in linear transformation of the centered observed mixtures, to obtain new vectors which are white. Fig.3. shows the implementation of whitening process. The components of a whitened vector are uncorrelated and their variances equals to unity. This means that the covariance matrix of whitened
data is equal to identity matrix. One way to perform whitening is using Eigen value Decomposition (EVD) method.

The whitening matrix is given by $P = ED^{-1/2} E^T$
where E is the orthogonal matrix of eigenvector found from the covariance matrix $E\{XX^T\}$
D is the diagonal matrix of the eigen values associated with each eigenvector.

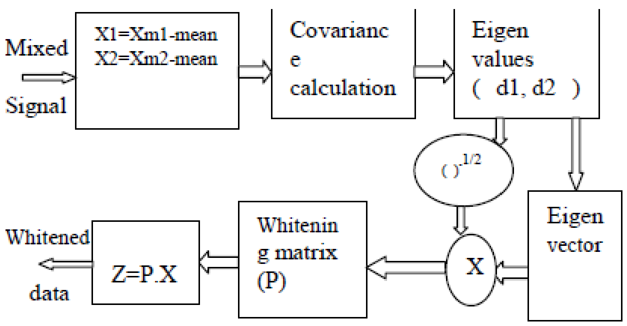

Fig.3. Implementation of whitening

IV. THE PROPOSED FASTICA ALGORITHM BASED ON MODIFIED SFLA

A Fast fixed-point algorithm for independent component analysis of complex valued signals was proposed by E.Bingham and A. Hyvarinen .Due to simplicity and fast convergence, Fast ICA is considered as One of the most popular solutions for linear ICA BSS problem .The algorithm involves the preprocessing as discussed in previous chapter and iteration scheme. For improving the performance of the algorithm, contrast function optimization is done in Fast ICA For efficiently using this algorithm over wide range of real values of signals, all the calculations are done in floating point system. An evolutionary optimization algorithm that mimics the social behavior of natural biological objects/species is an exciting development in optimization area. Several types of evolutionary computing methods are available in the literature. The Shuffled Frog Leap optimization Algorithm is method that mimics the memetic evolution of a group of frogs when seeking for the location that has maximum amount of food. Though this SFLOA ends up with local minima, Mutation and crossover operators are introduced to avoid getting trapped in local minima .It converges better in lesser time when compared to other optimization algorithms.

In this proposed algorithm, random vectors are assumed as frogs and frogs are seen as hosts for memes and are described as a memetic vectors. They can communicate with each other and improve their memes by passing information among each other which is mutation. When applying optimization technique, the entire population of ‘n’ frogs is divided into a number of frog memeplexes. The algorithm begins by randomly selecting F frogs and sorting them in descending order, according to their fitness value. Then, the frogs are divided into m memeplexes. For each memeplex, q frogs are selected to form a sub-memeplex. The division is done with the first frog going to the first memeplex, second one going tothe second memeplex, the pth frog to the pth memeplex and thep+ 1^{th} frog back to the first memeplex. The chance of being selected is proportional to the frog fitness; fitter frogs have a higher chance of being included in the sub memeplex.

A. *Iteration for one unit:*

The proposed fast ICA algorithm for one unit estimates one row of the demixing matrix as a vector that is an extreme of contrast functions. Fast ICA is an iterative fixed point algorithm, derived from contrast function. Assume Z is the whitened data vector and w^T is first two frogs after sorting is done. Estimation of new w^T or new frog or $w^T(k+1)$ is done iteratively with following steps until a convergence is achieved.
1) Choose initial frog of ‘n’ numbers at random.
2) Find norm of pair of frogs and divide by corresponding norms.
3) Update the frog by the formula W(k+1)_E{z $(w(k)Z^T)^3$}-3w(k) Where Z is whitened vector.
4) Calculate the fitness value from w(k+1)-w(k) and sort the frogs according to fitness value.

5) If w(k+1)-w(k) < ε is not satisfied for any one frog, then go back to step 2 by taking new 'w' as iniial one.ε is aconvergence parameter (~10-4)
6) When w(k+1)-w(k) < ε is satisfied for all the frogs, apply mutation for all the best frogs to produce new frogs.
7) Repeat from step 2 once.
8) Among '2n' frogs, frog that has good fitness value is seleced for demixing matix.

B. Fixed-point Iteration for finding Several ICs:

The independent components (ICs) can be estimated one by one using deflationary approach or can be estimated simultaneously by using symmetric approach. In the deflationary approach, it must be ensured that the rows of the separating matrix are orthogonal. In order to prevent that the algorithm estimates the same component more than one time, the following orthogonalization as in (5) is made .This verification is done by subtracting the projections of all previously estimated vectors before normalization from the current estimate after every iteration step.

V. RESULTS AND DISCUSSION

This proposed Independent component analysis algorithm is modeled in VHDL and implemented in FPGA using Xilinx 9.1i.Many ICA algorithms used for signal/image processing applications are slow in processes due to complicated arithmetic and time-consuming iterative computation. Bymaking use of VLSI technology, features such as high processing speed, hierarchy and modularity techniques are implemented .To overcome the complexity of ICA algorithms and to provide fast convergence, optimization is introduced to Fast ICA. Though FAST ICA provides better result, it doesn't provide weight vectors for accurate solution .So Fast ICA is Modified using optimization technique to overcome this problem. The simulation output of proposed ICA is shown in Fig 4.

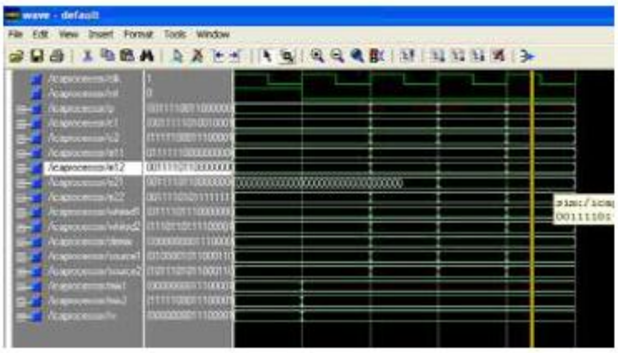

Fig 4.Simulated Output of Modified SFLA based ICA

VI. CONCLUSION

The hierarchy involves dividing an ICA process into sub processing modules until the complexity of the bottom sub modules becomes manageable. These sub modules are independently developed, then integrated together and put into a design and development environment for performing tasks such as synthesis, optimization, placement, and routing. The use of modularity enables the parallelism of the design process. Thus the use of modularity and hierarchy simplifies and speeds up the ICA process. The usage of optimization algorithm enables to find global optimal solution and also fast convergence. ICA algorithms are mostly applied in signal and image processing field, which usually entails large volumes data that are transferred in and out of the VLSI designs. A successful ICA hardware implementation that meets these requirements is possible with proposed method.

REFERENCES

[1].Hyvarinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," Neural Comput., vol. 9, no. 7, pp.1483–1492,Oct. 1997.

[2].Amit Acharyya and Koushik "hardware Efficient Fixed-Point VLSI Architecture for 2D Kurtotic FastICA"

[3] H. Du, H. Qi and X. Wang, "Comparative Study of VLSI Solutions to Independent Component Analysis", IEEE Trans. Industrial Electronics, vol. 54, no. 1, February, 2007.

[4] K. K. Shyu, M. H. Lee, Y. T. Wu and P. L. Lee, "Implementation of Pipelined FastICA on FPGA for Real-Time Blind Source Separation", IEEE Trans. Neural Networks, vol. 19, no. 6, pp. 958-970, June, 2008.

[5] Hyvarinen, "Fast and Robust Fixed-Point Algorithms for Independent Component Analysis",IEEE Trans. Neural Networks, vol. 10, no. 3, May, 1999.

[6] E.Bingham and A. Hyvarinen, A Fast fixed-point algorithm for independent component analysis of complex valued signals_, International Journal of Neural Systems, Vol. 10, No. 1, pp.1-8, February, 2000.

[7] Alan Paulo, Ana Maria,"FPGA hardware design, simulation and synthesis for a Independent component analysis algorithm using system-level design software.