Quantum Genetic Algorithm for Binary Decision Diagram Ordering Problem

Abdesslem Layeb and Djamel-Eddine Saidouni,

Lire laboratory, Vision&Infography group, University Mentouri of Constantine, Algeria

Summary
The Binary Decision Diagram (BDD) is used to represent in symbolic manner a set of states. It's largely used in the field of formal checking. The variable ordering is a very important step in the BDD optimization process. A good order of variables will reduce considerably the size of a BDD. Unfortunately, the search for the best variables ordering has been showed NP-difficult. In this article, we propose a new iterative approach called QGABDD based on a Quantum Genetic Algorithm. QGABDD is based on a basic core defined by a suitable quantum representation and an adapted quantum evolutionary dynamic. The obtained results are encouraging and attest the feasibility and the effectiveness of our approach. QGABDD is distinguished by a reduced population size and a reasonable number of iterations to find the best order, thanks to the principles of quantum computing

Key words:
Combinatorial problem, Quantum computing, Quantum Genetic Algorithm, Binary Decision Diagram,

1. Introduction

The objective of the checking application and electric circuits is to detect the errors which they contain or to show that they function well. One of the methods used in system checking is the model-checking [1]. One of the difficulties encountered in the domain of formal verification is the combinatorial explosion problem. For example in the model checking, the number of states in the transition graphs can reach prohibitive level, which makes their manipulation difficult or impossible. Consequently, compression methods are used in order to reduce the size of the state graph. The compression is done by using data structures in order to represent in a concise manner the set of states. In this case, the operations are done so on set of states rather than on explicit states.

The representation by the Binary Decision Diagrams BDD [2] is among the most known symbolic notations. The BDD is a data structure used to represent Boolean functions. The BDD is largely used in several fields since they offer a canonical representation and an easy manipulation. However, the BDD size depends on the selected variable order. Therefore it is important to find variable order which minimizes the number of nodes in a BDD. Unfortunately, this task is not easy considering the fact that there is an exponential number of possible variable ordering. Indeed, the problem of variable ordering was shown Np-difficult [3]. For that, several methods were proposed to find the best BDD variable order and which can be classified in two categories. The first class tries to extract the good order by inspecting the logical circuits [4], whereas, the second class is based on the dynamic optimization of a given order [5].

One of the iterative methods that have been developed recently to solve this type of problem is Genetic Algorithms GA. It is a stochastic iterative algorithm which maintains a population of individuals. GA adapts nature optimizing principles like mechanics of natural selection and natural genetics. Each individual represents a feasible solution in the problem search space. Basically, a genetic algorithm consists of three essential operations: selection, crossover, and mutation. The selection evaluates the fitness of each individual and keeps the best ones among them. The others are removed from the current population. The crossover merges two individuals to provide new ones. The operator of mutation allows moving each solution to one of its neighbours in order to maintain a good diversity during the process of optimization. GA allows guided search that samples the search space. Although GAs have been showed to be appropriate for solving BDD ordering problem [6], their computational cost seems to be a dissuasive factor for their use on large instances. To overcome this drawback and in order to get better speed and quality convergence, their implicit parallelism is exploited.

Quantum computing is a new research field that encompasses investigations on quantum mechanical computers and quantum algorithms [7]. QC relies on the principles of quantum mechanics like qubit representation and superposition of states. QC is capable of processing huge numbers of quantum states simultaneously in parallel. QC brings new philosophy to optimization due to its underlying concepts. Recently, a growing theoretical and practical interest is devoted to researches on merging evolutionary computation and quantum computing [8, 9]. The aim is to get benefit from quantum computing capabilities to enhance both efficiency and speed of classical evolutionary algorithms. This has led to the

Manuscript received September 5, 2007
Manuscript revised September 20, 2007

design of Quantum inspired Genetic Algorithms QGA that have been proven to be better than conventional GAs. Unlike pure quantum computing, QGA doesn't require the presence of a quantum machine to work.
In this context, we propose in this article, a new iterative approach called QGABDD based on QGA. For that, a problem formulation in terms of quantum representation and evolutionary dynamic borrowing quantum operators were defined. The quantum representation of the solutions allows the coding of all the potential orders with a certain probability. The optimization process consists in the application of a quantum dynamic constituted of a set of quantum operations such as interference, quantum mutation and measure. The experiences carried out on QGABDD showed the feasibility and the effectiveness of our approach.
Consequently, the remainder of the paper is organized as follows: section 2 presents some basis concepts of BDD. A brief introduction to quantum computing is presented in section 3. The proposed approach is described in section 4. Section 5 illustrates some experimental results. Then, we finish by giving conclusion and some perspective.

2. Binary Decision Diagram

A Binary Decision Diagram or BDD is data structure used for representation of Boolean functions in the form of rooted directed acyclic graph. It is composed of decision nodes and two final nodes called 0-final and 1-final (fig.1). The root and the intermediary nodes are indexed and possess two child nodes called high and low. BDD is called "ordered" if the different variables appear in the same order on all the ways from the root. It is important to note that for a given order of variables, the minimal binary decision graph is single. A BDD can be reduced while using the two following rules [2, 10, 11]:

- Recognize and share identical sub-trees.
- Erase nodes whose left and right child nodes are identical.

It is very important to take into account the order of variables to be used when using the BDD in practice. The size of a BDD is largely affected by the choice of the variable ordering (fig. 2). Unfortunately, there are an exponential number of possible orders (permutation). It is completely clear that the problem of variables ordering is NP-difficult. The use of heuristics is essential to find acceptable solutions within reasonable times. Within this perspective, we are interested in applying quantum computing principles to solve the variable ordering problem.

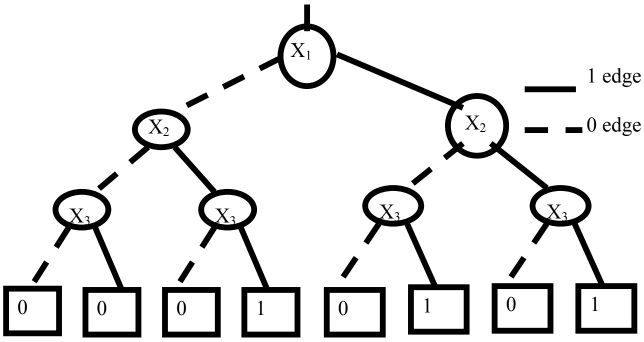

Fig1. Binary Decision Diagram for the Boolean function: $f = X1\ X3 + X2\ X3$

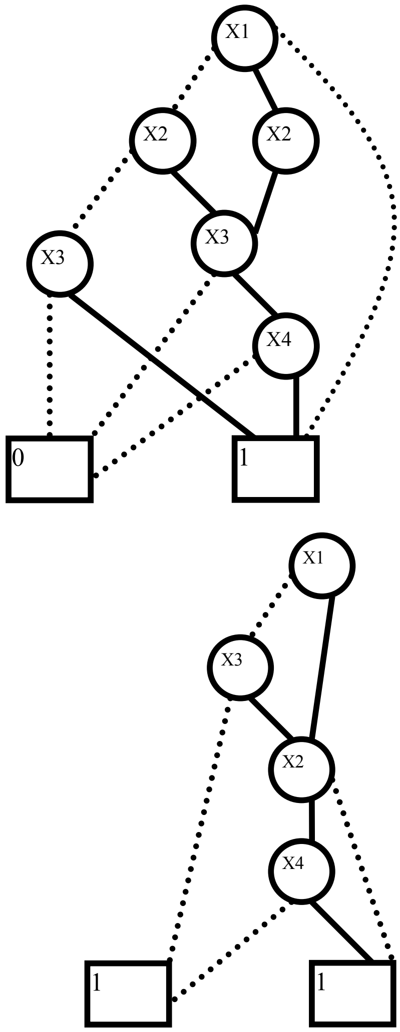

Fig.2. Two BDDs representation of the function: $(x1 \vee x3) \wedge (x2 \Rightarrow x4)$, in the left the order of variable is: x1, x2, x3, x4; right: order: x1, x3, x2, x4.

3. An Overview of Quantum Computing

Quantum Computing QC is an emergent field calling upon several specialties: physics, engineering, chemistry, computer science and mathematics. QC uses the specificities of quantum mechanics for the processing and the transformation of information. The aim of this integration of knowledge is the realization of a quantum computer in order to carry out certain calculations much more quickly than with a traditional computer. This acceleration is made possible while benefiting from the quantum phenomena such as the superposition of states, the entanglement and the interference. A particle according to principles of quantum mechanics can be in a superposition of states. By taking account of this idea, one can define a quantum bit or the qubit which can take value 0, 1 or a superposition of the two at the same time. Its state can be given by [8]:

$$\Psi = \alpha\,|0\rangle + b|1\rangle \qquad (1)$$

Where $|0\rangle$ and $|1\rangle$ represent the classical bit values 0 and 1 respectively; α and β are complex numbers such that

$$|\alpha|^2 + |b|^2 = 1 \qquad (2)$$

The probability that the qubit collapses towards 1 (0) is $|\alpha|^2$ ($|b|^2$).This idea of superposition makes it possible to represent an exponential set of states with a small number of qubits. According to the quantum laws like interference, the linearity of quantum operations and entanglement make the quantum computing more powerful than the classical machines. Each quantum operation will deal with all the states present within the superposition in parallel. For in-depth theoretical insights on quantum information theory, one can refer to [7].

A quantum algorithm consists in applying of a succession of quantum operations on quantum systems. Quantum operations are performed using quantum gates and quantum circuits. Yet, a powerful quantum machine is still under construction. By the time when a powerful quantum machine would be constructed, researches are conducted to get benefit from the quantum computing field. Since the late 1990s, merging quantum computation and evolutionary computation has been proven to be a productive issue when probing complex problems. Like any other GA, a Quantum Evolutionary Algorithm QGA [8, 9] relies on the representation of the individual, the evaluation function and the population dynamics. The particularity of QGA stems from the quantum representation they adopt which allows representing the superposition of all potential solutions for a given problem. It also stems from the quantum operators it uses to evolve the entire population through generations.

4. The Proposed Approach

The development of the suggested approach called QGABDD is based basically on a quantum representation of the research space associated with the problem and a quantum dynamic used to explore this space by operating on the quantum representation by using quantum operations.

4.2 Quantum representation of variable order

The problem of variable ordering can be mathematically formulated as follow:

Given a set of variables V={X_1, X_2… X_n}, the problem of BDD variables ordering can be defined by specifying implicitly a pair (Ω, SC) where Ω is the set of all possible solutions that is potentials variables order and *SC* is a mapping $\Omega \rightarrow R$ called score of the variable ordering. This score is the BDD size. Each solution is viewed as permutation of the V variables. Consequently, the problem consists to define the best permutation of V that gives the minimal BDD size.

In order to easily apply quantum principles on variable ordering, we need to map potential solutions into a quantum representation that could be easily manipulated by quantum operators. The variable order is represented as binary matrix satisfying the following criteria:

- For N variable, the size of the matrix is N*N. The columns represent the variables and the rows represent their order.
- The presence of 1 in the position *(i,j)* indicates that the rang of the variable *j* is *i* in the variable ordering.
- In each column and row there is a single 1.

The figure 3 shows the binary representation of the order {2, 1, 4, 3}, In terms of quantum computing, each order is represented as a quantum register as shown in figure 4. The register contains superposition of all possible permutations. Each column $\begin{pmatrix} a_i \\ b_i \end{pmatrix}$ represents a single qubit and corresponds to the binary digit 1 or 0. The probability amplitudes a_i and b_i are real values satisfying $|a_i|^2 + |b_i|^2 = 1$. For each qubit, a binary value is computed according to its probabilities $|a_i|^2$ and $|b_i|^2$. $|a_i|^2$ and $|b_i|^2$ are interpreted as the probabilities to have respectively 0 or 1. Consequently, all feasible variable orders can be represented by a quantum matrix *QM* (fig5) that contains the superposition of all possible variable permutations. This quantum matrix can be viewed as a probabilistic representation of all potential order. When embedded within an evolutionary framework, it plays the role of the chromosome. Only one chromosome is needed to represent the entire population.

$$\begin{matrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{matrix}$$

Fig. 3 Binary representation of the variable ordering

$$\left(\begin{array}{c|c|c|c} \mathbf{a}_1 & \mathbf{a}_2 & \multirow{2}{*}{...} & \mathbf{a}_m \\ \mathbf{b}_1 & \mathbf{b}_2 & & \mathbf{b}_m \end{array} \right)$$

Fig.4 Quantum register

$$\begin{bmatrix} \left(\begin{array}{c|c|c|c} \mathbf{a}_{11} & \mathbf{a}_{12} & \cdots & \mathbf{a}_{1m} \\ \mathbf{b}_{11} & \mathbf{b}_{12} & & \mathbf{b}_{1m} \end{array} \right) \\ \left(\begin{array}{c|c|c|c} \mathbf{a}_{21} & \mathbf{a}_{22} & \cdots & \mathbf{a}_{2m} \\ \mathbf{b}_{21} & \mathbf{b}_{22} & & \mathbf{b}_{2m} \end{array} \right) \\ \vdots \\ \left(\begin{array}{c|c|c|c} \mathbf{a}_{n1} & \mathbf{a}_{n2} & \cdots & \mathbf{a}_{nm} \\ \mathbf{b}_{n1} & \mathbf{b}_{n2} & & \mathbf{b}_{nm} \end{array} \right) \end{bmatrix}$$

Fig. 5 Quantum representation of variable ordering.

4.2 Quantum operators

The quantum operations used in our approach are as follows:

4.2.1. Measurement: This operation transforms by projection the quantum matrix into a binary matrix (fig.6). Therefore, there will be a solution among all the solutions present in the superposition. But contrary to the pure quantum theory, this measurement does not destroy the superposition. That has the advantage of preserving the superposition for the following iterations knowing that we operate on traditional machines. The binary values for a qubit are computed according to its probabilities $|a_i|^2$ and $|b_i|^2$. The binary matrix is then translated into a succession of integers.

4.2.2. The quantum interference: This operation amplifies the amplitude of the best solution and decreases the amplitudes of the bad ones. It primarily consists in moving the state of each qubit in the direction of the corresponding bit value in the best solution in progress. The operation of interference is useful to intensify research around the best solution. This operation can be accomplished by using a unit transformation which achieves a rotation whose angle is a function of the amplitudes a_i, b_i and of the value of the corresponding bit in the solution reference (fig 7). The values of the rotation angle δθ is chosen so that to avoid premature convergence. It is set experimentally and its direction is determined as a function of the values of a_i, b_i and the corresponding element's value in the binary matrix (table1).

$$\left(\begin{matrix} \begin{pmatrix} 0.99 \\ 0.14 \end{pmatrix} & \begin{pmatrix} 0.3778 \\ -0.9259 \end{pmatrix} & \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} & \begin{pmatrix} 0.8884 \\ -0.459 \end{pmatrix} \\ \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} & \begin{pmatrix} 0.3778 \\ -0.9259 \end{pmatrix} & \begin{pmatrix} 0.99 \\ 0.14 \end{pmatrix} & \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} \\ \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} & \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} & \begin{pmatrix} 0.3778 \\ -0.9259 \end{pmatrix} & \begin{pmatrix} 0.8884 \\ -0.459 \end{pmatrix} \\ \begin{pmatrix} 0.8884 \\ -0.459 \end{pmatrix} & \begin{pmatrix} 0.8884 \\ -0.459 \end{pmatrix} & \begin{pmatrix} 0.14 \\ 0.99 \end{pmatrix} & \begin{pmatrix} 0.99 \\ 0.14 \end{pmatrix} \end{matrix} \right) \xrightarrow{\textbf{Measure}} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Fig. 6 Quantum measurement.

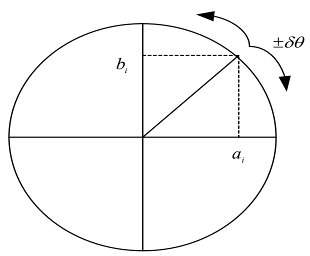

Fig. 7. Quantum interference

Table 1. Lookup table of the rotation angle

a	*b*	Reference bit value	Angle
> 0	> 0	1	+δθ
> 0	> 0	0	-δθ
> 0	< 0	1	-δθ
> 0	< 0	0	+δθ
< 0	> 0	1	-δθ
< 0	> 0	0	+δθ
< 0	< 0	1	+δθ
< 0	< 0	0	-δθ

4.2.3 *Mutation operator:* this operator performs permutation between two qubits (fig8). It allows moving from the current solution to one of its neighbors. It consists first in selecting randomly a register in the quantum matrix. Then, pairs of qubits are chosen randomly according to a defined probability. This operator allows exploring new solutions and thus enhances the diversification capabilities of the search process.

4.2.4 Crossover operators: Crossovers are important for promoting the exchange of high quality blocks within the population. They exchange subparts of two quantum chromosomes. For example the figure 9 shows a quantum crossover.

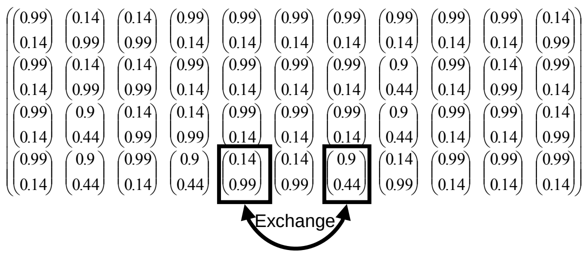

Fig8. Mutation operator

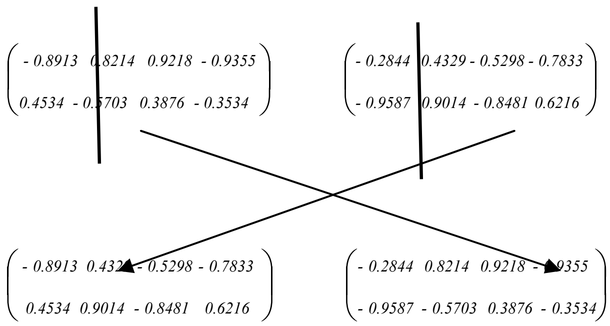

Figure 9. Quantum crossover

4.3 Outline of the proposed framework

Now, we describe how the representation scheme including quantum representation and quantum operators has been embedded within an evolutionary algorithm and resulted in a hybrid stochastic algorithm performing variable order search.

Given a set *S* of BDD variables to be ordered, first, a quantum matrix *QM(0)* is constructed to represent all possible orders. Starting from an initial binary matrix *BM(0)* extracted from the quantum matrix using the measurement operation, the algorithm progresses through a number of generations according to a quantum based dynamics. At each iteration, the following main tasks are performed: The assessment of the current order, the application of the interference operation, the application of the crossover and mutation operations, and the application of the measurement operation. The assessment of the current order intends to compute the fitness score of the solution. The evaluation of the solutions is done by using the size of the BDD obtained from the order as criterion of selection. In more details, the proposed QGABDD can be described as follow:

Input: A set of variable *ord*

(1) Construct the initial Population of Quantum Matrix *PQM*

(2) Generate an initial variable order ord_0, let *BM* be the corresponding binary matrix.

(3) Set $ord_{best} = ord'$ and $SC_{best} = SC(ord')$.

Repeat

(4) Apply an interference operation on *PQM* according to the best solution.

(5) Apply a crossover operation on *PQM*

(6) Apply a mutation operation on *PQM* according to the permutation probability pm.

(7) Apply a measurement operation on each chromosome to derive a new binary matrix BM_i.

(8) For veach BM_i Evaluate the corresponding order ord_i

(9) if $SC(ord_{best}) > SC(ord_i)$ then $ord_{best} = ord_i$ and $SC_{best} = SC(ord_i)$.

Until a termination-criterion is reached

Output: ord_{best} and $SC(ord_{best})$

5. Implementation and Evaluation

QGABDD is implemented in java 1.5 and is tested on a microcomputer with a processor of 2 GHZ and 256 MO of memory. We have used the package JBDD [12] which contains a set of tools for the creation and the manipulation of BDD. To assess the efficiency and accuracy of our approach several experiments were designed. The tests are divided into two classes. The first one contains Boolean functions built with the logical operations 'AND', 'XOR' and 'NOT' (table 2). However, the tests of the second class are Boolean functions containing the logical operations 'AND', 'OR' and 'NOT' (table 3). In all experiments, the size of the population is 4, the permutation probability is a tunable parameter which was set to 0.15, the interference angle is *π/20,* and the iteration numbers vary between 100 iterations for small tests and 1000 iterations for large tests. The found results are encouraging and prove the feasibility and the efficiency of our approach. Wilcoxon matched-pair signed-rank test were carried out to test the significance of the difference in the accuracy of our method. Indeed, at threshold *a=0.05*, there is significant difference between final solutions and initial solutions.

Table 2. Results given by QGABDD for the tests containing the logic operators: AND, XOR and NOT

Test	Number of variables	Initial Solution	Final Solution
Test1	100	276	151
Test2	100	299	145
Test3	120	330	183
Test4	120	355	190
Test5	130	331	209
Test6	135	381	206
Test7	135	419	211

Table 3. Results given by QGABDD for the tests containing the logic operators: AND, OR and NOT

Test	Number of variables	Initial Solution	Final Solution
Test1	90	132	91
Test2	95	152	98
Test3	100	166	103
Test4	110	170	113
Test5	120	180	124
Test6	130	205	134
Test7	135	225	139

6. Conclusion

In this work, we discuss the use of Quantum genetic algorithms to improve the variable ordering of a given BDD. The proposed approach called QGABDD is based on a quantum Genetic algorithm. The quantum representation of the solutions allows the coding of all the potential variable orders with a certain probability. The optimization process consists of the application of a quantum dynamics constituted of quantum operations such as the interference, the quantum mutation and measurement. The size of the population is considerably reduced thanks to the superposition principle. The experimental studies prove the feasibility and the effectiveness of our approach. As ongoing work we study the effect of crossover operation on the performance of our approach.

REFERENCE

[1]. E. M. Clarke, O. Grumberg, et D. E. Long. "Model checking and abstraction". ACM Transactions on Programming Languages and Systems, 16(5) :1512–1542, septembre 1994.

[2]. R. Drechsler, et B. Becker. "Binary Decision Diagrams: Theory and Implementation". Kluwer Academic Publisher, 1998.

[3]. B. Bollig and I. Wegener. "Improving the variable ordering of OBDDs is NPcomplete". IEEE Trans. on Comp., 45(9):993–1002, 1996.

[4]. H. Fujii, G. Ootomo, and C. Hori. Interleaving based variable ordering methods for ordered binary decision diagrams. In Int'l Conf. on CAD, pages 38–41, 1993.

[5]. N. Ishiura, H. Sawada, and S. Yajima. Minimization of binary decision diagrams based on exchange of variables. In Int'l Conf. on CAD, pages 472–475, 1991.

[6]. W. Lenders and C. Baier .Genetic Algorithms for the Variable Ordering Problem of Binary Decision Diagrams, Foundations of Genetic Algorithms, LNCS V. 3469/2005, pp. 1-20, June 2005

[7]. C.P. Williams and S.H. Clearwater. "Explorations in quantum computing". Springer Verlag, Berlin, Germany 1998.

[8]. K.H. HAN and J.H. KIM. "Quantum-inspired Evolutionary Algorithms with a New Termination Criterion, Hε Gate, and Two Phase Scheme," IEEE Transactions on Evolutionary Computation, IEEE Press, vol. 8, no. 2, pp. 156-169, April 2004.

[9]. A. Layeb, S. Meshoul and M. Batouhe.: "Multiple Sequence Alignment by Quantum Genetic Algorithm". In the 20th IPDPS 2006, ISBN: 1-4244-0054-6, pp 1 – 8, Greece, April 2006.

[10]. R. Drechsler, et B. Becker. Binary Decision Diagrams: Theory and Implementation. Kluwer Academic Publisher, 1998.

[11]. R. E. Bryant, "Symbolic Boolean Manipulation with Ordered Binary Decision Diagrams", ACM Computing Surveys, Vol. 24, No. 3, pp. 293-318, 1992.

[12]. J. Whaley, "JAVABDD, a Java Binary Decision Diagram Library", Stanford University http://javabdd.sourceforge.net/.

Abdesslem Layeb: PHD student. In 2005, he received his M.S. degree in informatics from the University Mentouri of Constantine, Algeria. Member of LIRE laboratory in the University of Constantine, his recent work is dealing with the combinatorial optimization problems in Formal methods.

Dr. Djamel-Eddine Saidouni: in 1996, he received his PHD degree in informatics from the university Paul Sabatier of Toulouse, France. Lecturer and senior researcher, LIRE laboratory Vision&Infograpghy group, University Mentouri of Constantine, Algeria.