

DIFFERENTIAL EVOLUTION (DE) ALGORITHM TO OPTIMIZE BERKELEY - MAC PROTOCOL FOR WIRELESS SENSOR NETWORK (WSN)

¹ALAA KAMAL, ²M.N. MOHD. WARIP, ³MOHAMED ELSHAIKH
, ⁴ R.BADLISHAH

^{1,2,3,4} Embedded Network and Advanced Computing Research Cluster, School of Computer and Communication engineering, *University Malaysia Perlis, Level 1, Pauh Putra Main Campus, 02600, Arau, Malaysia*

E-mail: ¹kamalyousif@studentmail.unimap.edu.my, ²nazriwarip@unimap.edu.my, ³elshaikh@unimap.edu.my, ⁴badli@unimap.edu.my

ABSTRACT

The Energy efficient being the main constraint for wireless sensor networks (WSN) became an important area of interest for various researches in WSN. A lot of work on conservation power consumption at a different layer of protocols is present till date, of which energy conservation at a Medium access control (MAC) layer is catching a lot of attention. Generally, The energy consumption it is a main critical factor affecting the operational lifetime of individual nodes and the entire network. The energy consumption in medium access control (MAC) for wireless sensor network (WSN) affected by many reasons collision, overhearing, Idle Listening, Control Overhead. The optimizing power consumption is a challenge to prolong lifetime network. In this paper, we minimize the power consumption of the B-MAC protocol for wireless sensor network (WSN) by using Differential Evolution (DE) algorithm. Simulation experiments are carried out on the discrete event simulator OMNET++ for the purposes of this research paper. Moreover, the experimental analysis shows the improvement the energy consumption of (DE-BMAC) can be conserved about (58.6%) more than B-MAC without affect the network performance.

Keywords: *Berkeley MAC Protocol, Differential Evaluation (DE), Optimization, Power Consumption, WSN*

1. INTRODUCTION

MAC protocols draw a lot of attention due to their effect on the lifetime of wireless Sensor Network (WSN). Sensor nodes are driven by batteries which have a limited lifetime. Also, these nodes are deployed in the harsh environment where their change or recharge becomes difficult. An energy efficient design and implementation of MAC protocol can reduce the battery consumption and increase the sensor nodes lifetime. Traditional MAC protocols are cannot solve the energy efficient for WSN .So, we need modification in MAC protocols to minimize the power consumption and maximum Network performance[1]. In Berkeley-MAC protocols there many parameters can affect the power consumption and overall performance. Some parameters are listed for B-MAC protocol, slot duration, Check interval, Bit rate, max transmission power. In literature, there has an amount of researches MAC

protocols in WSN network. Recently, there are an amount of researches in MAC protocols used different optimization technique in designs and implantations. In 2015, B. Sharma et al. [2] using a Genetic algorithm(GA) to improvement Performance in terms of increased PDR and Throughput and reduced End to End Delay with an optimized-OLSR (OLSR-GA) protocol. K. V. Patil et al., in 2013 [3] presented by Using Genetic algorithm (GA) algorithm a new automatic selection of optimal configuration can offer more enhanced performance. OLSR routing protocol the standard greedy approach is replaced with necessity first algorithm. Using proposed protocol the network traffic load of an administrative packet is reduced. The proposed routing protocols are best suitable for a vehicular network which is highly dynamic in nature. In 2012, J. Toutouh et al. [4] studied to reduce the energy consumption of the OLSR routing protocol in VANETs by using Differential Evolution algorithm to search for



energy-efficient configurations. The experimental analysis shows that significant improvements over the standard configuration can be attained in terms of energy savings (up to 30%) without degrading the QoS. In 2011, J. Toutouh et al. [5] Assess between OLSR and DE-OLSR by using IEEE 802.11p definition and considering different urban areas sizes, traffic densities, and workloads. The QoS has been measured using four metrics: PDR, NRL, E2ED, and RPL. This comprehensive performance evaluation shows that DE-OLSR is better-suited for VANETs than the standard version, improving its resource consumption and scalability. J. Toutouh et al. [6] Deals with the optimal parameter setting of the OLSR, a well-known mobile ad hoc network routing protocol, by defining an optimization problem. Using Differential Evolution (DE), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Simulated Annealing (SA) algorithms in order to find automatically optimal configurations of this routing protocol. In the literature for designing and implementation of routing protocols that aimed to reduce the energy consumption, however, a few works are dedicated to optimizing and analyzing the affect the parameters MAC protocols set on the energy consumption for wireless sensor network (WSN). In This paper we define an optimization power consumption to the Berkeley-MAC (B-MAC) protocol, obtaining the configuration that with a specific characteristic of wireless sensor network (WSN). Furthermore, the evaluated done by using Differential Evaluation (DE) algorithm. Chosen Differential Evaluation (DE) algorithm because is a simple and presented wide solvers for Real-valued parameter. Simulation experiments are done to optimize and analyze B-MAC protocol parameters set against energy consumption, throughput, and packet delivery ratio (PDR). Moreover, concern to minimize the power consumption, and maximize the Throughput and packet delivery ratio (PDR). The paper is organized as follow: section 2 gives an overview of Differential Evaluation (DE). Section 3 describes the proposed framework of this work. Section 4 describe the Experimental setting. The Result and Discussion in section 5 and we conclude this paper in Section 6.

2 METHODOLOGY

Differential Evolution (DE) has recently emerged as a simple and efficient algorithm for global optimization over continuous spaces. DE it is much easier to implement and applies a kind of

differential mutation operator on parent chromosomes to generate the offspring. Since its inception in 1995, DE has pulled the attention of many researchers all over the world, resulting in a lot of variant of the basic algorithm, with improved performance[7]:

2.1 Differential Evolution Algorithm (DE)

DE is a simple and easy evolutionary algorithm. DE it works through steps a simple cycle of stages, Below we explain each step separately[8][9]

2.1.1 INITIALIZATION

DE searches for an optimum point in a D-dimensional continuous space. It begins with a randomly initiated population of NP D – dimensional Real-valued parameter vectors. Each vector, also known as the genome, forms a candidate solution to the multidimensional optimization problem. Appear subsequent generations in DE by discrete time steps like $t = 0, 1, 2 \dots, t+1$ the successive generations are represented by $G, G+1, G+2, \dots, g, g+1$. The parameter vectors are likely to be changed over different generations, we adopt the following notation for representing the i -th vector of the population at the current generation (at time $t = t$) as:

$$X_i(t) = [x_{i,1}(t), x_{i,2}(t) \dots x_{i,D}(t)]$$

Where

$$i = 1, 2, \dots, NP.$$

At the very starting of a DE run at $t = 0$, problem parameters or independent variables are initialized somewhere in their convenient numerical range. So, if the j -th parameter of the given problem has its lower and upper bound as $X_{min,j}$, $X_{max,j}$, and $rand_{i,j}(0, 1)$ indicated the j -th instantiation of a uniformly distributed random number between 0 and 1 for i -th vector, then we may initialize the j -th component of i -th population members as:

$$\vec{X}_{i,j}(0) = X_{min,j} + rand_{i,j}(0, 1) * (X_{max,j} - X_{min,j})$$

2.1.2 MUTATION

Mutation mean (biology) change a genome that has characteristics resulting from chromosomal alteration. In the evolutionary computing paradigm, mutation is also seen as a change with a random element. DE, applies not to generate increments, but to randomly sample vector differences like

$$\Delta \vec{X}_{r_2, r_3} = (\vec{X}_{r_2} - \vec{X}_{r_3})$$

In DE, mutation amounts to creating a donor vector $\vec{V}_i(t)$ for changing each population member $\vec{X}_i(t)$, in each generation (or in one iteration of the algorithm). To create $\vec{V}_i(t)$ for each i-th member of the current population (also called the target vector), three other distinct parameter

Vectors say the vectors $\vec{X}_{r_1}^i$, $\vec{X}_{r_2}^i$, and $\vec{X}_{r_3}^i$ are chosen randomly from the current population. Where r_1^i , r_2^i , and r_3^i are mutually exclusive integers randomly chosen from the range [1, NP], which are also different from the base vector index i. These indices are randomly generated once for each mutant vector. Now the difference of any two of these three vectors are scaled by a scalar number F and the scaled difference is added to the third one whence we obtain the donor Vector $\vec{V}_i(t)$, can express the process as,

$$\vec{V}_i(t) = X_{r_1}^i(t) + F (X_{r_2}^i(t) - X_{r_3}^i(t)) \tag{1}$$

The mutation scheme has different kinds of DE schemes but in this research used the formula in Equation (1).

2.1.3 CROSSOVER

To increase the potential diversity of the population, a crossover operation comes into play after generating the donor vector through mutation. The DE family of algorithms can use two kinds of crossover schemes - exponential and binomial. The donor vector exchanges its body parts.

Components with the target vector $X_i(t)$ under this operation to form the trial vector

$$U_i(t) = [u_{i,1}(t), u_{i,2}(t) \dots u_{i,D}(t)]$$

Binomial crossover is performed on each of the D variables whenever a randomly choose number between 0 and 1 is less than or equal to the CR value. The scheme may be outlined as:

$$\vec{U}_{i,j,G} = \begin{cases} \vec{V}_{i,j,G} & , \text{ If } (\text{rand } i, j (0,1) \leq \text{CR}) \\ \vec{X}_{i,j,G} & , \text{ otherwise} \end{cases}$$

2.1.4 SELECTION

The last phase of a DE-iteration is the “selection”. Deciding who between the target vector $\vec{X}_i(t)$ and the newly formed trial vector $\vec{U}_i(t)$ will survive to the next generation. The decision whether original $X_i(t)$ will be retained in the population or will be replaced by $U_i(t)$ in the next time step t+1 is entirely dependent upon the ‘survival of the fittest’ concept. If the trial vector yields a better fitness value it will replace the target vector in the next time step. Here by better fitness value we mean a lower value of the objective function in case of a minimization problem, and a higher value of the same if it is a maximization problem. The selection operation may be outlined as:

$$\vec{X}_i(t+1) = \begin{cases} \vec{U}_i(t) & \text{if } F(\vec{U}_i(t)) \leq F(\vec{X}_i(t)) \\ \vec{X}_i(t) & \text{if } F(\vec{U}_i(t)) > F(\vec{X}_i(t)) \end{cases}$$

Where F(X) is the function to be minimized. Any one between the target vector and its offspring survives the population size remains fixed throughout generations. The fitness of the population members either improves over generations or remains unchanged, but never deteriorates.

The selection mechanism of the classical DE algorithm. The objective function we are trying to minimize is the three dimensional sphere function given by,

$$F(x) = \sum_{i=1}^n X^2 \tag{2}$$

Deferential Evaluation (DE) Algorithm Steps

- Step 1:** Select NP individuals $X_{i,j,g}$ randomly.
- Step 2:** For $i = 1$ to NP let $F_i = F(X_{i,j,g})$
- Step 3:** While (convergence criterion not yet met) do steps 4 to 10
- Step 4:** For $i = 1$ to NP do steps 5 to 10
- Step 5:** Select three different random indexes r_0, r_1 and r_2 between 1 to NP ($i \neq r_0 \neq r_1 \neq r_2$)
- Step 6:** $\vec{V}_{i,j,g} = \vec{X}_{r_0,j,g} + F (\vec{X}_{r_1,j,g} - \vec{X}_{r_2,j,g})$
- Step 7:** For $j = 1$ to n do steps 8 to 9
- Step 8:** Select randomly r_j variable ($0 \leq r_j < 1$) and j_{rand} index ($1 \leq j_{rand} \leq n$)
- Step 9:** If $r_j < CR$ then $U_{j,l,g} = V_{j,l,g}$ else $U_{j,l,g} = X_{j,l,g}$
- Step 10:** If $F(U_{i,j}) \leq F_i(x_{i,j})$ then $X_{i,g+1} = U_{i,g}$ else $X_{i,g+1} = X_{i,g}$

3 PROPOSED FRAMEWORK

The optimization of the Berkeley – MAC protocol parameters configuration is carried out through two phases, optimization phase, and simulation phase. Figure 1 explains the main components of proposed framework for DE- Berkeley MAC protocol for WSN. Generally, the framework is consist of optimizing algorithm and evaluation procedure. Differential Evolution (DE) algorithm is used as an intelligent technique to find optimal values.

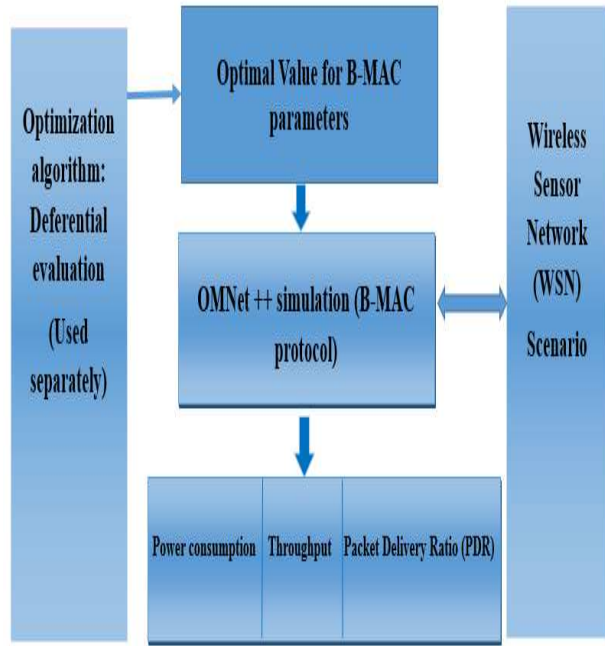


Figure 1: Optimization framework for DE-BMAC Protocol

An optimal vector of a real variable range is produced, and each one represents a new defined B-MAC parameters and fitness function. the fitness function for DE is defined to give a specific weight performance to power consumption, Throughput, and Packet delivery ratio (PDR) shown Equation 3, Equation 4, and Equation 5 respectively to calculate the simulation result.

$$\text{Power consumption} = \sum_{i=1}^n \text{power consume by hosts} \tag{3}$$

$$\text{Throughput (bit)} = \frac{\sum \text{Number of packet receive} * 8}{\text{time simulation}} \tag{4}$$

$$\text{PDR} = \frac{\sum \text{Number of packet receive}}{\sum \text{Number of packet send}} \tag{5}$$

4 EXPERIMENTAL SETTING

The parameters and values used to calculate the DE-BMAC explain in Table 1. Additionally, Table 2 define the DE algorithm parameters for the optimization process.

Table1: DE-BMAC Configuration

Main parameters	Range of values
Slot duration (s)	[0.0025,0.00125, 0.0006]
Bit rate (bps)	[4915200, 9830400, 19660800]
Check interval (s)	[0.001 0.002, 0.004]
Max power transmission (mw)	[1, 2, 4]

Table2: Parameters Of DE Optimization Algorithm

Parameters	value
Crossover probability (CR)	0.9
Mutation Factor (F)	0.9

According to, the steps of DE algorithm defined previously the optimum values explain in Table 3

Table 3: Optimum Values For DE-BMAC Protocol

Parameters	Optimum Value
Slot Duration	0.00107 s
Bit Rate	34406400bps
Check interval	0.007s
Tx power	7 mw

The simulation was carried out by using OMNET++ simulator over defined wireless sensor network (WSN) scenario has been carried out based on B-MAC parameters, range values for DE-BMAC, and WSN parameters explain in Table 4

Table 4: Main Parameters Defined For OMNET++ Simulation

Factors	Values
Energy Type	Ideal energy storage
Sensor Nodes	21 host
Constrain Area	200×200 m ²
Mobility Type	Stationary mobility, circle mobility
MAC Protocol	Berkeley – MAC
Radio Medium Type	APSK ScalarRadio Medium

5 RESULT AND DISCUSSION

The optimum value for DE-BMAC protocol carried out by using OMNET++ simulator and evaluated the result with B-MAC protocol.

Figure 2 illustrate power consumption versus the different message length used the B-MAC and DE-BMAC. It is seen the DE-BMAC outperformed the B-MAC protocol in term minimizes power consumption. However, Power consumption increased when the increased the size of message length.

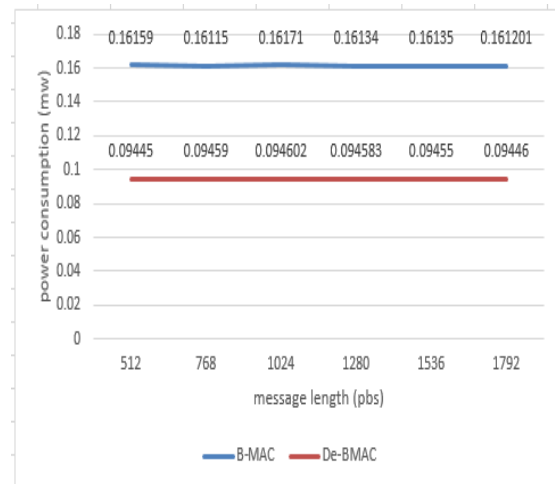


Figure 2: DE-BMAC Power Consumption

Figure 3 explain the Throughput of B-MAC protocol and DE-BMAC. In general, the performance evaluation shows that the Throughput

increased with the DE-BMAC more than B-MAC protocol. However, Throughput decreased when the increased the size of message length.

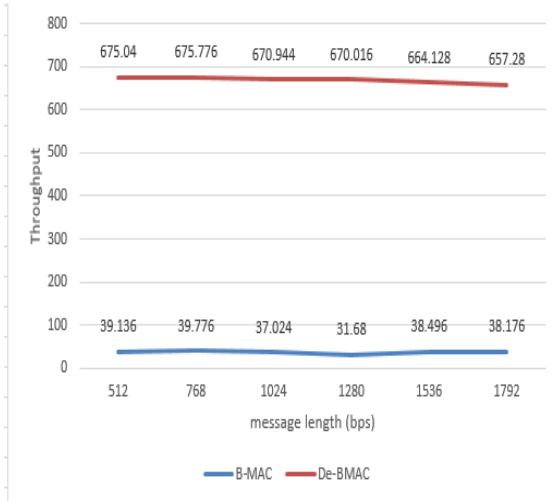


Figure 3: De-Bmac Throughput

The evaluation of Packet Delivery Ratio (PDR) for DE-BMAC is shown Figure 4. Can see the DE-BMAC achieved higher PDR than B-MAC protocol.

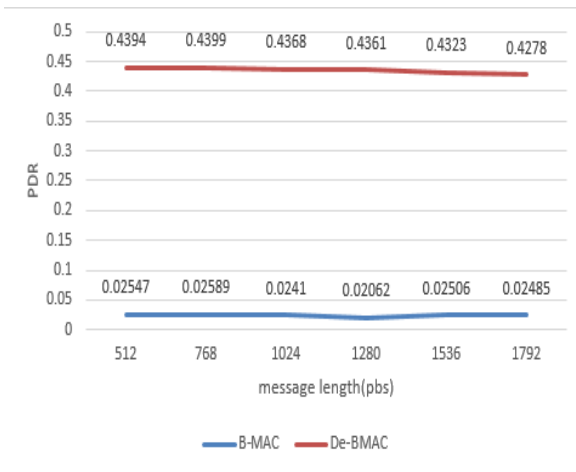


Figure 4: DE-BMAC PDR

6 CONCLUSION

Throughout this paper presented optimization process to Berkeley-MAC (B-MAC) protocol,

named DE-BMAC, the B-MAC parameters using the Differential Evaluation (DE) algorithm for wireless Sensor Network (WSN). The DE allows determining the optimal value for the B-MAC parameters in WSN scenario. Overall, we can say that by employing Differential Evaluation (DE) in DE-BMAC framework, managed to optimize the parameters selection.

REFERENCES:

- [1] R. Kaur and M. Lal, "Wireless Sensor Networks: A Survey," *Int. J. Electron. Commun. Technol.*, vol. 7109, pp. 102–106, 2013.
- [2] B. Sharma, "Simulation and Automatic Tuning of OLSR Routing Protocol for VANETs," vol. 5, no. 5, pp. 644–648, 2015.
- [3] K. V. Patil and M. R. Dhage, "The Adaptive Optimized Routing Protocol for Vehicular Ad - hoc Networks," vol. 2, no. 3, pp. 67–70, 2013.
- [4] J. Toutouh and E. Alba, "Green OLSR in VANETs with Differential Evolution," *Proc. 14th Annu. Conf. Companion Genet. Evol. Comput.*, no. Rfc 3626, pp. 11–18, 2012.
- [5] J. Toutouh and E. Alba, "Optimizing OLSR in VANETS with differential evolution," *Proc. ACM DIVANet '11*, no. January 2011, p. 1, 2011.
- [6] J. Toutouh, "Intelligent AODB Routing Protocol Optimization for," pp. 1–10.
- [7] A. I. Technology, "A DIFFERENTIAL EVOLUTION ALGORITHM PARALLEL," vol. 86, no. 2, pp. 184–195, 2016.
- [8] A. W. Mohamed, H. Z. Sabry, and M. Khorshid, "An alternative differential evolution algorithm for global optimization," *J. Adv. Res.*, vol. 3, no. 2, pp. 149–165, 2012.
- [9] W. Gong, Z. Cai, and C. X. Ling, "DE/BBO: A hybrid differential evolution with biogeography-based optimization for global numerical optimization," *Soft Comput.*, vol. 15, no. 4, pp. 645–665, 2011.