CHAPTER 5

ENHANCED AOMDV ROUTING WITH HYBRID BAT OPTIMIZATION ALGORITHM

5.1 INTRODUCTION

Harmony Search (HS) (Geem et al. 2001) refers to a new metaheuristic method for minimizing potentially non-differentiable as well as non-linear functions in continuous spaces. HS owes its inspiration to the behavior of a musician’s improvisation procedure, wherein all musicians attempt to enhance the current tune for creating optimum harmony in real-world musical performances. HS protocols originated on similar lines between engineering optimizations as well as musical improvisations. Engineers look for globally best solutions as defined by objective functions, similar to how music professionals look for aesthetic harmony as defined by the aesthetician. When improvising music, all musicians choose pitches in feasible ranges, yielding one harmony vector. If every pitch yields excellent solution, the experience is retained in the parameter’s memory, and the potential of yielding improved solution increases subsequently. Moreover, the novel method needs small number of control variables that ensures that HS is easy for implementation, more robust, as well as being extremely adequate for parallel computations.

BAT is a powerful protocol in exploitation (local search), at times trapped into local optimum, thereby being unable to carry out global searches well. As BAT algorithms depend on arbitrary walks for solutions search, and
fast convergence is not BAT guaranteed. To increase population diversity to avoid being trapped in local optima is hybridized with HS. An improvement is made in BAT by adding pitch adjustment to HS, which serves as a mutation operator to speed up convergence (Yang 2010).

Additionally, fine adjustment of variables $\alpha$ and $\gamma$ which are constants used for updating the loudness and pulse rate emission impact the convergence rates of the bat protocol. Particularly, variable $\alpha$ behaves in a like fashion as the cooling schedule in the simulated annealing. Although the execution is more complex than other metaheuristic protocols, it shows that it uses a balanced combination of the benefits of already present successful protocols with novel features on the basis of echolocation behavior of bats (Yang 2010). Fresh solutions are created through the adjustment of frequencies, loudness as well as pulse emission rate and the acceptance of presented solution relies on the quality of solutions qualified by loudness as well as pulse rates that are connected to the nearness for fitness of location or solution to the global optimum solution.

A related expansion is the usage of several schemes of wavelength or frequency variants rather than current linear execution. Additionally, the rate of pulse emission as well as loudness may also be different in a more intricate fashion. A further expansion for discrete issues is the usage of time delay between pulse emission as well as echoes bounced back. For instance, in the travelling salesman issue, the distance between two adjacent nodes or cities may be encoded as time delay. As micro bats utilize time difference between their two ears for obtaining 3D information, they are capable of identifying kind of prey as well as velocity of flying insects. Hence, an additional expansion to the present BAT algorithm is the usage of directional echolocation as well as Doppler Effect that might result in more intriguing variations as well as novel protocols.
Optimization algorithms may be split into two primary groups: deterministic as well as stochastic algorithms. The former use gradients like hill climbing to make rigorous moves while also generating identical sets of solutions when iterations begin with same origin point. (Wang & Guo 2013). Stochastic optimization method have its basis in real-valued evolutionary algorithms and is utilized for exhaustive exploration of design parameter space when looking for best link (Sedano et al. 2012).

Hybrid optimizations work on the assumption that a combination of two or more protocols has been implemented for one optimization problem. Hybrid optimizations use heuristic approach for choosing the best of the algorithms to employ in a particular setting. Here to generate hybrid register allocators which choose between two distinct register allocation models, graph coloring as well as linear scan (Cavazos et al. 2006). The aim is the creation of allocators which achieve excellent balance between two factors: one is attempting to discover excellent packing of parameters to registers and therein attaining excellent runtime performance and another attempting to decrease allocator overheads.

Deterministic method utilizes a local optimization method such as EA for improving efficacy through reduction of extreme CPU time. The deterministic method is executed in EA in two phases. The first is fitness evaluation wherein deterministic method is utilized for obtaining efficient new error predictors. In the latter phase, deterministic method refines the solutions yielded by the evolutionary component of the protocol. The novel error predictor allows for the evolution of several individuals in all generations, obviating the discarding of well-adapted linkages which other mechanisms do not discover.
The difference between HS/BAT and BAT is that mutation operator improves the solution of BAT algorithm thus this method explores novel search space by mutation of HS/BAT algorithm and exploits population data available with BAT thereby avoiding falling into local optima in BAT (Lenin et al. 2014).

5.2 METHODOLOGY

BAT is a powerful protocol in exploitations (local searches), at times trapped in local optimum, thereby being unable to carry out global searches well. For bat algorithm, searches depend on arbitrary walks and so rapid convergence is not guaranteed. The bat algorithm variants available in the literature are Fuzzy Logic Bat Algorithm (FLBA) Khan et al. (2011), Multiobjective bat algorithm (MOBA) (Yang 2011), K-Means Bat Algorithm (KMBA) (Komarasamy & Wahi 2012), Chaotic Bat Algorithm (CBA) (Lin et al. 2012), Binary bat algorithm (BBA) (Nakamura et al. 2012), Differential Operator and Levy flights Bat Algorithm (DLBA) (Xie et al. 2013).

In this work, to increase population diversity for BAT to avoid being trapped in local optima, it is hybridized with Harmony Search (HS). An improvement is made in BAT by adding pitch adjustment to HS which serves as a mutation operator to speed up convergence. A hybrid BAT (HBAT) is proposed. The difference between HS/BAT and BAT is that mutation operator improves the solution of BAT algorithm. Thus, this method explores new search spaces by mutating HS/BAT algorithms and exploits population data available with BAT thereby avoiding falling into local optima in BAT.

5.2.1 Harmony Search (HS)

HS is a new metaheuristic optimization protocol inspired by natural musical performance procedures which occur when musicians are searching for optimal harmony states. HS algorithm optimization operator is delineated as Harmony Memory (HM) that maintains solution vectors within search
space. Harmony memory Size (HMS) that represent solution vectors retained in HM; Harmony Memory Consideration Rate (HMCR) which represent replay any well-known piece of music (a set of pitches in harmony), the Pitch Adjustment Rate (PAR) play something identical to a known piece in player’s memory (thereby altering the pitch slightly); the pitch adjustment (BW) play completely novel or arbitrary pitch from possible ranges.

HS algorithm has many advantages over other meta-heuristic algorithms considered above: (a) HS algorithms impose lesser mathematical requisites and do not need initial value settings of the decision parameters. (b) As HS algorithms use stochastic arbitrary search, derivative data is also not necessary (c) HS algorithms generate new vectors after taking into consideration all present vectors while GA merely regards two parent vectors. These characteristics improve flexibility of HS algorithms and yield improved solutions (Patil & Patel 2013).

HS was inspired by the improvisation of Jazz musicians who individually refine their individual improvisation through a piece of music variation resulting in an aesthetic harmony. Stages of HS protocol are:

**Begin**

**Step 1** : Initialize the HM.

**Step 2** : Evaluate the fitness.

**Step 3** : While the termination criterion is not satisfied do

for d = 1:D do

if rand < HMCR then // memory consideration

xnew (d) = xa (d) where a ∈ (1, 2, . . . ,HMS)

if rand < PAR then // pitch adjustment
\begin{align*}
\text{xnew}(d) &= \text{old}(d) + bw \times (2 \times \text{rand} - 1) \\
\text{endif} \\
\text{else} & \quad // \text{arbitrary selection} \\
\text{xnew}(d) &= \text{min}, + \text{rand} \times (\text{max}, \text{min} - \text{min} - d) \\
\text{endif} \\
\text{endfor } d \\
\text{Update the HM as } xw = \text{xnew}, \text{ if } (\text{xnew}) < (xw) \text{ (minimization objective)} \\
\text{Update the best harmony vector} \\
\text{Step 4} &: \quad \text{end while} \\
\text{Step 5} &: \quad \text{Output results.} \\
\text{End} \\
\text{The steps of the protocol are detailed below:} \\
\text{Stage 1. Setting up of the optimization issue as well as algorithm variables:} \text{ In the initial stage, the optimization issue is given by:} \\
\text{Minimize } (\text{or Maximize}) f(x) \\
\text{subject to } x_i \in X_i, \ i = 1,2,\ldots, N. \\
\text{Minimize } (\text{or Maximize}) f(x) \\
\text{subject to } x_i \in X_i, \ i = 1,2,\ldots, N. \\
\text{The solution is obtained by either maximizing or minimizing as required. In this work, for a specified source to destination, several paths are identified through the link measure in AOMDV y modifying path discovery}
procedure. For maintaining a balance between network loads as well as QoS, the aim is the minimization of packet loss rate, estimated load, and the delay in route.

F(i) are scalar objective functions to be optimized. $X_i$ refers to the set of potential range of values of all decision variables $x_i$ (continuous decision variables).

Additionally, the control variables of HS are also given in this particular stage. The variables are the HMS that is the quantity of solution vectors (population members) in the HM in every generation; HMCR; PAR; and the Number of Improvisations (NI) or terminating criteria (Chakraborty et al. 2009 and Amiri et al. 2010).

**Stage 2. HM initialization:** In this stage, every element of every vector in the parallel population of HM that is the size of HMS is set with a uniformly distributed arbitrary number between upper as well as lower bound.

$$[L^i, U^i], \text{wherein } 1 \leq i \leq N.$$ This is performed for the i-th element of the j-th solution vector using the following equation:

$$x_i^j = L^i + \text{rand}(0,1).(U^i - L^i)$$

wherein $j = 1, 2\ldots$. HMS and rand $(0, 1)$ is a uniformly distributed arbitrary number between 0 and 1 and it is sampled anew for every element of every vector.

**Stage 3. New Harmony improvisation:**

In this step, a novel Harmony vector $\vec{x} = (x_1, x_2, x_3, \ldots, x_N)$ is created on the basis of three rules:
(1) Memory consideration,
(2) Pitch adjustments, and
(3) Arbitrary selection.

Creating a novel harmony known as ‘improvisation’ will take the form:

\[ x_i \leftarrow \begin{cases} 
    x_i \in \{x_i^1, x_i^2, x_i^3, \ldots, x_i^{HMS} \} & \text{with probability HMCR} \\
    x_i \in X_i & \text{with probability } (1 - \text{HMCR}) 
\end{cases} \]

This step helps to explore the solution space for better solutions.

**Stage 4. HM update:** If the new Harmony vector \( \breve{x} = (x_1, x_2, x_3, \ldots, x_N) \) is better than the worst harmony in the HM, assessed with regard to the objective function value, the New Harmony is incorporated in the HM. This is actually the selection stage of the algorithm wherein the objective function value is tested for determining whether the new variant ought to be incorporated in the population (HM) or not.

The route formed by the solution is checked whether it is optimal or not based on the objective function.

**Stage 5. Check terminating criteria:** Last stage is to check the criterion for continuing the process the terminating criteria (maximal NI) is fulfilled, computations are stopped. Else, stages 3 and 4 are iterated.

Figure 5.1 shows the flowchart for HS.
Figure 5.1 Flowchart of HSA
5.2.2 Implementation of Hybrid Hs-Bat

A hybrid metaheuristic protocol through the induction of pitch adjustment in HS as a mutation operator in BAT protocol optimizes. The variation between HS/BAT as well as BAT is that mutation operators improve original BAT creating novel solutions for every bat. Thus, this technique explores fresh search spaces through mutation of HS protocols and exploits population data with BAT thereby avoiding falling into local optima in BAT (Lenin et al. 2014).

In proposed BAT-HS, the critical operator is the hybrid HS mutation operator that improvises harmony in HS with BA. The notion of the new hybrid mutation operator has its basis in two conventions. First, poor solutions absorb several novel used attributes from excellent ones. Second, mutation operators improve explorations of novel search spaces. In this manner, the original HS strong exploration capacities and exploitation abilities of BAT are fully developed.

As searches rely completely on arbitrary walks, rapid convergence is not capable of being ensured in bat algorithm. An improvement by appending mutation operator is made to BAT proposed in chapter 4, including three minor improvements which aim to speed up convergence, ensuring that the method is practical for applications, but with no loss of the original method’s attractive attributes.

The flowchart of the suggested Hybrid BAT (HBAT) is given in Figure 5.2.
Figure 5.2 Flowchart of the HBAT Algorithm
5.3 IMPLEMENTATION AND RESULTS

Opnet 14 was integrated with Matlab 2010 and simulations were carried out using node with a transmission range of 250m. The BAT parameters are the same as used in the previous chapter. Table 5.1 shows the summary of results. Figure 5.3 to 5.5 shows the average end to end delay, average jitter and average PDR respectively.

Table 5.1 Summary of Results

<table>
<thead>
<tr>
<th>Node pause time in Second</th>
<th>AOMDV</th>
<th>LQ-AOMDV</th>
<th>BAT-AOMDV</th>
<th>HBAT-AOMDV</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average end to end delay</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.8712</td>
<td>0.9245</td>
<td>0.8268</td>
<td>0.7706</td>
</tr>
<tr>
<td>25</td>
<td>0.5107</td>
<td>0.4711</td>
<td>0.485</td>
<td>0.4687</td>
</tr>
<tr>
<td>50</td>
<td>0.4035</td>
<td>0.3318</td>
<td>0.377</td>
<td>0.3598</td>
</tr>
<tr>
<td>75</td>
<td>0.2815</td>
<td>0.3076</td>
<td>0.2606</td>
<td>0.246</td>
</tr>
<tr>
<td>100</td>
<td>0.207</td>
<td>0.265</td>
<td>0.1928</td>
<td>0.1836</td>
</tr>
<tr>
<td><strong>Average Jitter in Second</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.0039</td>
<td>0.0041</td>
<td>0.0044</td>
<td>0.0041</td>
</tr>
<tr>
<td>25</td>
<td>0.0022</td>
<td>0.0011</td>
<td>0.0025</td>
<td>0.0024</td>
</tr>
<tr>
<td>50</td>
<td>0.002</td>
<td>0.0016</td>
<td>0.0013</td>
<td>0.0012</td>
</tr>
<tr>
<td>75</td>
<td>0.0008</td>
<td>0.0012</td>
<td>0.0006</td>
<td>0.0005</td>
</tr>
<tr>
<td>100</td>
<td>0.0011</td>
<td>0.0008</td>
<td>0.0008</td>
<td>0.0007</td>
</tr>
<tr>
<td><strong>Packet Delivery Ratio</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.7195</td>
<td>0.7139</td>
<td>0.728</td>
<td>0.7542</td>
</tr>
<tr>
<td>25</td>
<td>0.7597</td>
<td>0.8617</td>
<td>0.8871</td>
<td>0.9171</td>
</tr>
<tr>
<td>50</td>
<td>0.7852</td>
<td>0.8741</td>
<td>0.8889</td>
<td>0.9043</td>
</tr>
<tr>
<td>75</td>
<td>0.8442</td>
<td>0.9509</td>
<td>0.9657</td>
<td>0.9712</td>
</tr>
<tr>
<td>100</td>
<td>0.9191</td>
<td>0.9643</td>
<td>0.978</td>
<td>0.9817</td>
</tr>
</tbody>
</table>
From table 5.1 and figure 5.3 it is observed that for node pause time of 0 the average end to end delay of HBAT-AOMDV is reduced by 18.16% than LQ-AOMDV and by 7.04% than BAT-AOMDV. For node pause time of 75 the average end to end delay of HBAT-AOMDV is reduced by 22.25% than LQ-AOMDV and by 5.46% than BAT-AOMDV. For node pause time of 100 the average end to end delay of HBAT-AOMDV is reduced by 36.29% than LQ-AOMDV and by 4.89% than BAT-AOMDV.

Figure 5.4 Average Jitter
From Table 5.1 and Figure 5.4 it is observed that for node pause time of 0, Jitter of HBAT-AOMDV is reduced by 0.15% than LQ-AOMDV and by 6.63% than BAT-AOMDV. For node pause time of 75, Jitter of HBAT-AOMDV is reduced by 76.24% than LQ-AOMDV and by 4.12% than BAT-AOMDV. For node pause time of 100, Jitter of HBAT-AOMDV is reduced by 11.16% than LQ-AOMDV and by 7.14% than BAT-AOMDV.

Figure 5.5 Average Packet Delivery Ratio

From Table 5.1 and Figure 5.5 it is observed that for node pause time of 0, Packet Delivery Ratio of HBAT-AOMDV improves by 5.49% than LQ-AOMDV and by 3.54% than BAT-AOMDV. For node pause time of 75, Packet Delivery Ratio of HBAT-AOMDV improves by 2.11% than LQ-AOMDV and by 0.57% than BAT-AOMDV. For node pause time of 100, Packet Delivery Ratio of HBAT-AOMDV improves by 1.79% than LQ-AOMDV and by 0.38% than BAT-AOMDV.
From Figure 5.6 it is observed that the routing control overhead of proposed HBAT-AOMDV is increased when contrasted with AOMDV and decreased when contrasted with LQ-AOMDV and BAT-AOMDV. Average values of HBAT-AOMDV is increased by 10.71% than AOMDV and but decreased by 19.84% & 8.13% than LQ AOMDV and BAT-AOMDV.

Figure 5.7 Total Number of Route Discovery
In Figure 5.7 it is seen that the HBAT-AOMDV for a total number of route discovery gets decreased by the average of 12.52% & 18.53% when contrasted with AOMDV and LQ-AOMDV and but increased by 5.23% than BAT-AOMDV.

![Figure 5.8 Routing/Data Bits](image)

From Figure 5.8 it is seen that the HBAT-AOMDV for routing/Data bits gets decreased by the average of 8.45%-30% as opposed to AOMDV, LQ-AOMDV and BAT-AOMDV.

5.4 SUMMARY

For bat algorithm, search relies on arbitrary walks and so rapid convergence is not guaranteed. To increase population diversity for BAT to avoid being trapped in local optima, it is hybridized with Harmony Search (HS). An improvement is made in BAT by adding pitch adjustment to HS which serves as a mutation operator to speed up global convergence rate. The proposed hybrid H-BAT achieves better convergence than Bat algorithm and new meta hybrid approach to solve optimization.
Experimental results show that the average PDR of HBAT-AOMDV is improved than LQ-AOMDV and BAT-AOMDV. For node pause time of 100, Packet Delivery Ratio of HBAT-AOMDV improves by 1.79% than LQ-AOMDV and by 0.38% than BAT-AOMDV. Jitter and end to end delay of HBAT_AOMDV reduces than LQ-AOMDV and BAT-AOMDV. For node pause time of 100, Jitter of HBAT-AOMDV is reduced by 11.16% than LQ-AOMDV and by 7.14% than BAT-AOMDV. For node pause time of 100 the average end to end delay of HBAT-AOMDV is reduced by 36.29% than LQ-AOMDV and by 4.89% than BAT-AOMDV.