Neuro-fuzzy Architecture for Handover Decision Making in Emerging Heterogeneous Networks

Saida Driouache[1], Najib Naja[2], and Abdellah Jamali[3]

ABSTRACT: In emerging heterogeneous networks, seamless vertical handover is a critical issue. There must be a trade-off between the handover decision delay and accuracy. This paper's concern is to contribute to reliable vertical handover decision making that makes a trade-off between complexity and effectiveness. So, the paper proposes a neuro-fuzzy architecture that combines the capacity of learning of the artificial neural networks with the power of linguistic interpretation of the fuzzy logic. The architecture can learn from experience how executing a handover to a particular access network affects the quality of service. Vertical Handover Intelligent Control (VHIC) provides not only flexibility for initial deployment but also the adaptive capability to optimize the vertical handover with minimal human interference. Simulation results reveal that VHIC is fast, reliable, and enhances the throughput. Also, It decreases the end-to-end delay from near 3.5ms to 0.1ms, the jitter to almost 0μs, and the packet loss to 0%.

Keywords: Heterogeneous Network, Seamless Vertical Handover, Artificial Neural Network, Fuzzy Logic, Reinforcement Learning, Quality of Service

DOI: 10.37936/ecti-cit.2021153.242440

Article history: received October 1, 2020; revised November 21, 2020; accepted January 22, 2021; available online November 8, 2021

1. INTRODUCTION

Heterogeneous Networks (HetNets) have become necessary to overcome the limitations of wireless access networks, to help them to evolve, and to achieve the always-best-connected concept [1]. However, many challenges arise from the heterogeneity in mobile networks. Vertical Handover (VH) is still a significant challenge that needs resolution before a truly converged network can be relized. User Equipment (UE) executes a VH when it switches between two access networks of different technologies. VH is used to prevent a possible severe loss of performance or connection loss caused by technical reasons.

The advanced HetNets require fast, efficient, and reliable VH. Many existing handover approaches are motivated by their simplicity. However, most of them focus on either low complexity, reliability, or performance.

Artificial intelligence is required to keep up with the rapid evolution of future generations of HetNets and services. Learning neuro-fuzzy architectures are ever-evolving and can address the problem of uncertain, stochastic, and unavailable access network parameters in realistic HetNets. This paper proposes a reliable architecture to reduce handover decision delay and maintain the quality of real-time services at the highest level. The proposed architecture depends on Fuzzy Logic (FL), Artificial Neural Networks (ANNs), and Reinforcement Learning (RL). It learns from the experience of how executing a VH to a particular Candidate Network (CN) affects the Quality of Service (QoS).

The structure of the rest of this paper is as follows. Section 2 presents related work. Section 3 presents the proposed architecture. Section 4 explains the learning approach used. Section 5 presents a performance evaluation of the proposed architecture. Finally, Section 6 concludes the paper.

2. RELATED WORK

Section 1 mentions that many existing VH proposals focus on either complexity [2][3], performance

[1,2]The authors are with CEDOC2TI, INPT, Avenue Allal Al Fassi, Rabat, Morocco, E-mail: driouache@inpt.ac.ma, saida.driouache@gmail.com, naja@inpt.ac.ma

[3]The author is with IR2M laboratory, ENSA, Hassan 1^{st} University, BP 539, Route de Casa, Settat, Morocco, E-mail: abdellah.jamali@uhp.ac.ma

[4][5][6], or reliability [7]. In the paper [2], Ben Zineb et al. proposed a fuzzy Multiple Attribute Decision Making (MADM) approach to reduce the high VH decision delay and complexity of MADM solutions. The proposal in [3] reduces the handover delay significantly. This proposal integrates IEEE 802.21 media independent handover and software-defined networking. The MADM approach [4] satisfies the performance and user preferences to a greater extent. However, the VH decision delay is high. Also, users often choose a low-cost access network, which leads to handover failure in many instances. In paper [5], a two-stage FL based VH is developed based on the QoS-requirements. The simulation results show that it offers better performance than traditional MADM. The reference [6] presents a load-balancing VH approach between WiMAX and Wi-Fi. The system improves the capacity and QoS based on the information exchange between networks and mobile UEs. However, the exchange of information and the VH which are designed to give preference to users lead to a bottleneck in wireless networks. That further leads to handover failure. Li et al. [7] proposed Bandit and Threshold Tuning (BaTT) to minimize the regret of handover failures and augment the reliability in extreme mobility. BaTT uses ϵ-binary-search and further devises opportunistic Thompson sampling, which optimizes the sequence of the target CNs to measure for reliable handover with a particular regret. Their experiment over a real LTE dataset from Chinese high-speed rails validates a significant reduction of a 29.1% handover failure.

VH over emerging HetNets requires automation and intelligence [8]. AI approaches can achieve excellent performance because they are inspired by nature findings and human reasoning.

FL relies on human intelligence to simplify the decision-making process under uncertainty [9]. It takes fuzzy variables as inputs and rapidly produces a decision using IF-THEN rules. Therefore, FL has simple computation, implementation, and interpretation. Reference [10] presents a VH approach that maximizes the user's satisfaction. Simulation results validate that this approach can solve the ping-pong phenomenon and effectively take into account different scenarios, vehicle speeds, and network conditions. Authors in [11] used FL to filter out CNs that do not satisfy the QoS requirements. Thus, the processing time decreases. Authors in [12] used FL to reduce the ping-pong effect. But, the scalability of a FL controller in terms of input and output variables is extremely low because the configuration of the rules is manual. So, the FL implementation considers only a few decision criteria. Also, FL based proposals fail to adapt based on the application type.

Similar to the biological neural networks, ANNs are composed of connected nodes (neurons). ANNs afford a high level of learning from old decisions. In [13], the authors showed the ability of ANNs to provide automatic VH decision making. Besides, ANNs can achieve reliable performance within challenging scenarios [14][15]. When it comes to real-time applications, ANNs afford higher accuracy and efficiency. They tend to experience lower packet loss and satisfy users in terms of QoS. ANN learning/training increases VH decision delay. But they proceed in parallel, which permits solutions to problems where multiple constraints have to be satisfied simultaneously.

Taking note of the learning and reasoning abilities of the FL and ANNs approaches, the purpose of this paper proposal is to enhance performance and reliability and achieve seamless near real-time VH.

3. PROPOSED VERTICAL HANDOVER INTELLIGENT CONTROL ARCHITECTURE

The proposed Vertical Handover Intelligent Control (VHIC) architecture is conceptually similar to the Approximate Reasoning based Intelligent Control (GARIC) architecture proposed in [16]. In GARIC, the first ANN states evaluations, the second ANN applies FL to recommend actions, and the third ANN determines to what degree these actions should be modified. GARIC architecture tunes its controller through updating weights on links in these networks. As long as the learning proceeds, the followed action more often is the one recommended by the controller. Many real-world control problems can make use of GARIC; this paper proposes VHIC as an adapted version of it to control VHDM in HetNets. GARIC inspired our VHIC design for many reasons:

- The non-linearity between inputs and outputs makes the analytical modeling of the system very difficult.
- Approximate reasoning based controllers do not require analytical models.
- GARIC can learn and tune the controller even when only weak reinforcement (binary failure signal) is available. It provides an adaptive and non-linear system that discovers how to deliver flexibility in the input-output relation.
- GARIC uses a discounted reward approach that is less complicated than average reward approaches.

Figure 1 shows the proposed VHIC, where the controller and evaluator are a Decision Making Network (DMN) and an Evaluation Network (EN), respectively. DMN decides whether to continue with the currently connected network or to switch to another one. EN learns to become a good evaluator of DMN decisions. In GARIC, the controller and evaluator have the same inputs. In VHIC, EN inputs are the criteria values of the current access network (previous decision). Also, EN uses a boolean failure signal to indicate whether or not a failure occurs. The collection of decision criteria values for all available CNs is the input to DMN.

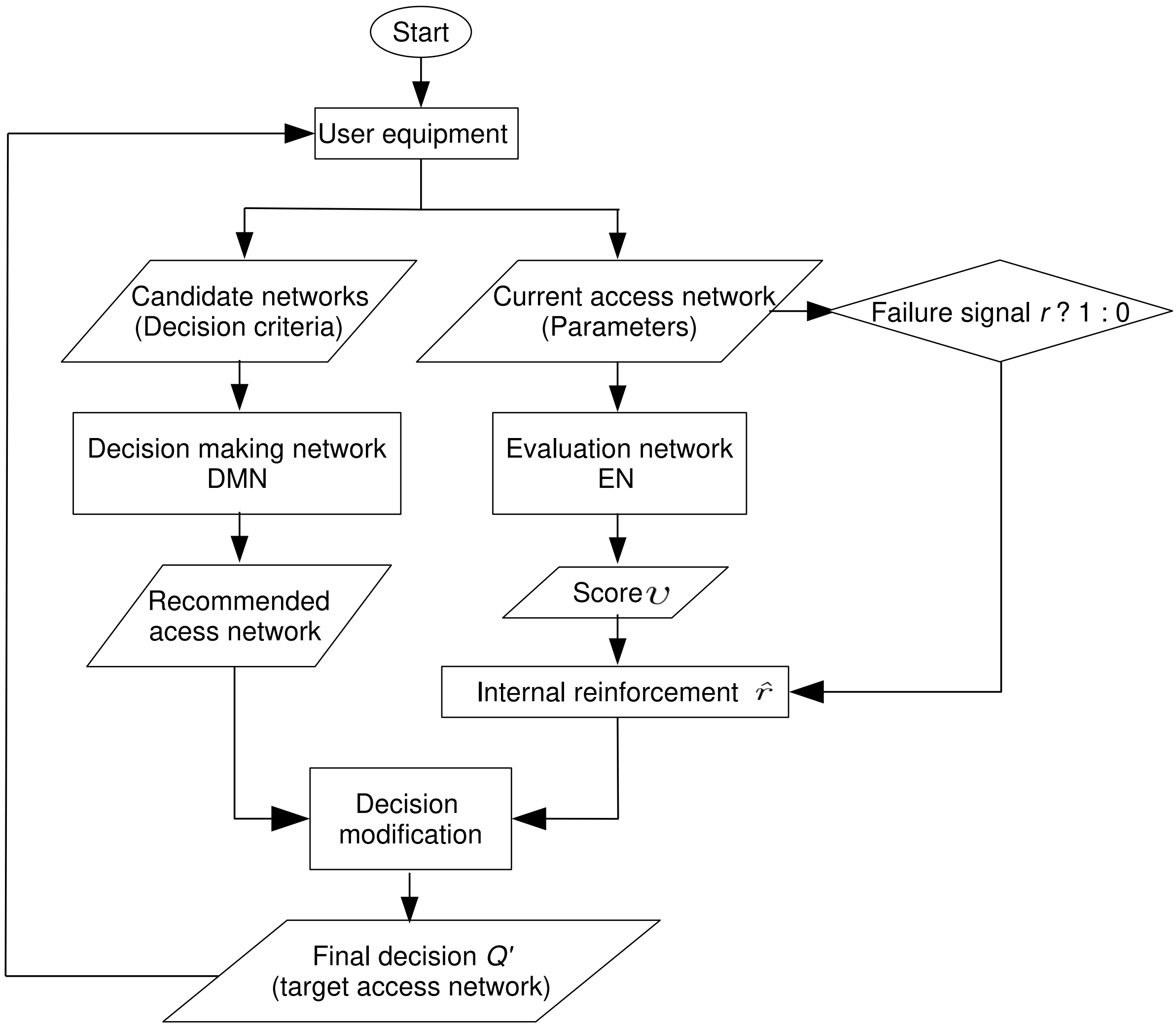

Fig.1: *Flowchart of VHIC Architecture*

3.1 Evaluation network

EN is a standard two-layer feed-forward network. This model is more compact and faster to evaluate than other ANNs. EN has two input nodes with a bias node, three hidden nodes, and one output node. As shown in Figure 2, A is the weights matrix of connections between input and hidden nodes. B is the weights matrix of connections between input and output nodes. C is the weights matrix of connections between hidden and output nodes. Each hidden node receives three inputs x_j and has three weights, while the output node has six weights and receives from hidden nodes and directly from input nodes. The output of hidden units y_i is the sigmoid activation function in Equation 1. The latter is preferred since it is real-valued, differentiable, and allows less computation.

$$y_i[t,t+1] = \frac{1}{1+exp(-\sum_{j=1}^{3} a_{ij}[t]x_j[t+1])} \quad (1)$$

In Equation 2, υ denotes EN output, which is a score indicating the decision goodness.

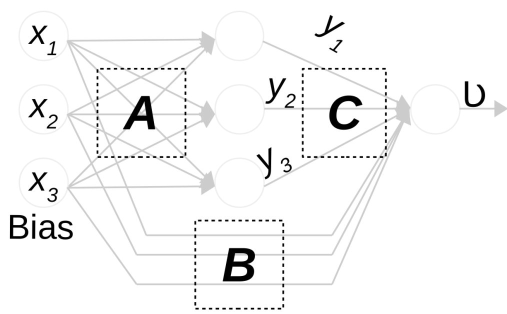

Fig.2: *Evaluation network*

$$v[t, t+1] = \sum_{j=1}^{3} b_j[t]x_j[t+1] + \sum_{i=1}^{3} c_i[t]y_i[t, t+1] \quad (2)$$

Using double-time dependencies t and $t+1$ detects whether weights update or the change in inputs cause the change in v, compares v over time, and notices whether better decisions or worse decisions are made. To predict the future reinforcement $\hat{r}$, Equation 3 suitably discounts v and combines it with the external boolean failure signal r. The latter indicates whether or not a decision failure occurs.

$$\hat{r}[t+1] = \begin{cases} 0 & start, \\ r[t+1] - v[t, t] & failure, \\ r[t+1] + \gamma v[t, t+1] - v[t, t] & otherwise \end{cases} \quad (3)$$

γ is the discount rate that tells how important future rewards are to the current state. γ takes a value between 0 and 1. The estimation above of $\hat{r}$ gives less weight to the future value of v than its current value.

3.2 Decision making network

DMN is a feed-forward neuro-fuzzy network with five layers of nodes. Each layer performs one stage of the fuzzy inference process (refer to Figure 3).

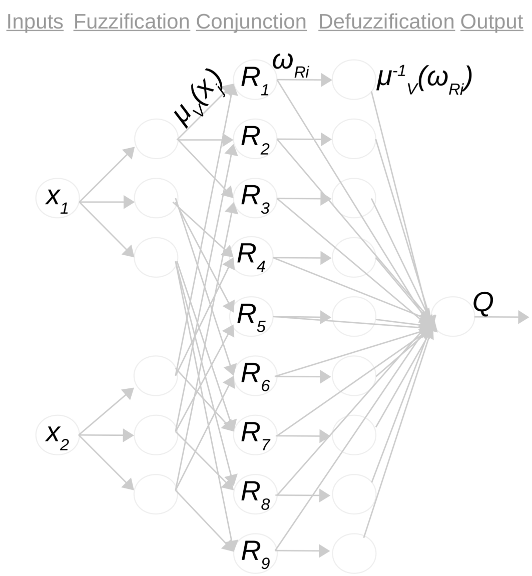

Fig.3: *Decision making network*

• Layer 1 is the input layer and consists of real-valued decision criteria. Its nodes do no computation. Many input variables may cause VHIC to be less sensitive, and handover might not occur when necessary. Therefore, there are only two input variables.

• Layer 2 is the fuzzification layer where a node corresponds to one possible linguistic value V of an input variable. A node feeds its output to all the rules that use the label V in their 'IF' part. This output is a Membership Function (MF) $\mu_V(x_j)$ that can take values ranging from zero to one. In this paper, VHIC uses triangular and trapezoidal MFs because of their simple formulas and computational efficiency. Besides, both triangular and trapezoidal MFs have been used extensively, especially in real-time implementations. The triangular function in Equation 4 is defined by a lower limit a, an upper limit b, and a value m, where $a < m < b$.

$$\mu_V(x) = \begin{cases} 0 & x \leqslant a \quad or \quad x \geqslant b, \\ \frac{x-a}{m-a} & a < x \leqslant m, \\ \frac{b-x}{b-m} & m < x < b \end{cases} \quad (4)$$

The trapezoidal MF is defined by a lower limit a, an upper limit d, a lower support limit b, and an upper support limit c, where $a < b < c < d$. Two special cases of trapezoidal MF are used; the first one is used with parameters $a = b$ (refer to Equation 5), and the second one is used with parameters $d = c$ (refer to Equation 6)

$$\mu_V(x) = \begin{cases} 0 & x > d, \\ \frac{d-x}{d-c} & c \leqslant x \leqslant d, \\ 1 & x < c \end{cases} \quad (5)$$

$$\mu_V(x) = \begin{cases} 0 & x < a, \\ \frac{x-a}{b-a} & a \leqslant x \leqslant b, \\ 1 & x > b \end{cases} \quad (6)$$

• Layer 3 implements the conjunction ω_R of rule R antecedents. Each node corresponds to a rule in the rule base. The inputs come from all the nodes in Layer 2 that participate in the 'IF' part of that rule. The conjunction is done by the Softmin operator in Equation 7 that is a softer version of the minimum operator.

$$\omega_R = \frac{\sum_i \mu_i exp(-\kappa\mu_i)}{\sum_i exp(-\kappa\mu_i)} \quad (7)$$

Parameter κ controls the Softmin operator hardness. We recover the usual minimum when $\kappa \longrightarrow \infty$.

• Layer 4 corresponds to consequents. The input of a consequent node comes from the rule that uses it. Its output is the inverse $\mu_V^{-1}(\omega_R)$, which means a defuzzification applied to the rule R. DMN applies the Local Mean of Maximum (LMoM) before the combination of consequents to determine $\mu_V^{-1}(\omega_R)$. LMoM works by inverting the non-constant portions. For a triangular MF, LMoM gives Equation 8.

$$\mu_V^{-1}(\omega_R) = m\omega_R + \frac{1}{2}(a+b)(1-\omega_R) \quad (8)$$

Equation 9 and 10 calculates LMoM for trapezoidal

MFs with parameters $a = b$ and $c = d$, respectively.

$$\mu_V^{-1}(\omega_R) = d + \omega_R(c - d) \quad (9)$$

$$\mu_V^{-1}(\omega_R) = a + \omega_R(b - a) \quad (10)$$

- Layer 5 has one node that combines recommendations from all fuzzy control rules in the rule base and uses the weighted sum in Equation 11 to calculate the output Q_{CN} for every CN. Then, DMN recommends the CN with maximum value Q.

$$Q_{CN} = \frac{\sum_R \omega_R \mu^{-1}(\omega_R)}{\sum_R \omega_R}, \qquad Q = \max Q_{CN} \quad (11)$$

3.3 Recommended decision modification

Values of $\hat{r}$ from the previous time step and the recommended decision Q are used to generate a final decision *Q'* that corresponds to the selected CN. *Q'* is stochastically chosen. The deviation function between Q and *Q'* must be some non-negative, monotonically decreasing function. For example, in this paper, the deviation cannot be larger than the $exp(-\hat{r})$ function 12.

$$|Q' - Q| \leq exp(-\hat{r}(t-1)) \quad (12)$$

The magnitude of this deviation is large when $\hat{r}$ is low and small when $\hat{r}$ is high. That means if the previously made decision is bad, there must be a large deviation from Q and an appropriate recommendation from DMN. In this case, there may be more than one CN to satisfy Equation 12. So, VHIC considers $Q' = Q_{CN}$ where $|Q_{CN} - Q|$ is the closest to the $exp(-\hat{r}(t-1))$ value. But, if the previously selected decision is a good one, the deviation is small. Then, possibly there is no other CN (with value Q_{CN}) to satisfy Equation 12. In this case, VHIC selects the recommended CN. That shows that DMN remains consistent with the fuzzy control rules.

4. LEARNING MECHANISM

4.1 Learning in evaluation network

In EN, learning occurs by adjusting weights and resembles the reward/punishment scheme in ANNs. If positive (negative) $\hat{r}$ is received, values of weights would be rewarded (punished) and changed in the direction that increases (decreases) their contribution to υ. Equation 13 updates the weights matrix B where β is the learning rate. Similarly, Equation 14 updates weights matrix C. The introduction of the bias is to fluctuate the activation function y_i to the right or left, and it has a constant value of 1. So, the bias term is not trainable.

$$b_j[t+1] = b_j[t] + \beta\hat{r}[t+1]x_j[t], \quad \beta > 0 \quad (13)$$

$$c_i[t+1] = c_i[t] + \beta\hat{r}[t+1]y_i[t,t] \quad (14)$$

Equation 15 updates weights matrix A. $\hat{r}$ is an error measure because no direct error measurement is possible, and knowledge of the correct decision is not available. Therefore, if $\hat{r}$ is positive, EN adjusts the weights to increase the score υ, and vice versa.

$$\begin{aligned} a_{ij}[t+1] = a_{ij}[t] + \\ \beta\hat{r}[t+1]y_i[t,t](1 - y_i[t,t])sgn(c_i[t])x_j[t] \end{aligned} \quad (15)$$

4.2 Learning in decision making network

DMN learning occurs by changing parameters that describe MFs and moving them to the left and right. It is sufficient to adjust only consequent MFs, since the modification of consequents may correct errors in the specification of antecedents. For this reason and computation simplicity, VHIC tunes only consequent MFs. DMN fixes all other weights at 1. Vector ρ includes parameters of all consequent MFs.

Q computation intends to maximize υ so that the system avoids failure. Therefore, gradient descent uses the learning rule in Equation 16 and estimates the derivative $\partial\upsilon/\partial\rho$, in order to maximize υ as the objective function of ρ.

$$\Delta\rho(t) = \eta s(t)\hat{r}(t)\frac{\partial\upsilon}{\partial Q}\frac{\partial Q}{\partial\rho} \quad (16)$$

Equation 17 calculates $\partial\upsilon/\partial Q$ by the instantaneous difference ratio because the dependence of υ on Q is quite indirect and computationally complex. The instantaneous difference ratio is not a good estimator of this derivative. Therefore, VHIC uses only its sign and assumes the inherent existence of the derivative.

$$\frac{\partial\upsilon}{\partial Q} \approx sgn\left(\frac{d\upsilon}{dQ}\right) \approx sgn\left(\frac{\upsilon(t) - \upsilon(t-1)}{Q(t) - Q(t-1)}\right) \quad (17)$$

Q is known, differentiable, and depends directly on ρ. Label V is parametrized by vector ρ_V which includes parameters a, b, m, and a, b, c, d of triangular and trapezoidal MFs, respectively. The derivatives in Equations 18, 19, and 20 are computed by substituting for μ_V^{-1} using Equations 8, 9, and 10, and differentiating using Equation 11. These derivatives are combined to compute $\partial Q/\partial\rho_V$.

- Triangular MF:

$$\begin{cases} \frac{\partial Q}{\partial a} = \frac{\partial Q}{\partial b} = \frac{1}{2}\frac{\omega_R}{\sum_r \omega_R}(1 - \omega_R), \\ \frac{\partial Q}{\partial m} = \frac{\omega_R^2}{\sum_r \omega_R} \end{cases} \quad (18)$$

- Trapezoidal MF with $a = b$:

$$\begin{cases} \frac{\partial Q}{\partial a} = \frac{\partial Q}{\partial b} = 0, \\ \frac{\partial Q}{\partial c} = \frac{\omega_R^2}{\sum_r \omega_R}, \\ \frac{\partial Q}{\partial d} = \frac{\omega_R}{\sum_r \omega_R}(1-\omega_R) \end{cases} \tag{19}$$

- Trapezoidal MF with $c = d$:

$$\begin{cases} \frac{\partial Q}{\partial a} = \frac{\omega_R}{\sum_r \omega_R}(1-\omega_R) \\ \frac{\partial Q}{\partial b} = \frac{\omega_R^2}{\sum_r \omega_R}, \\ \frac{\partial Q}{\partial c} = \frac{\partial Q}{\partial d} = 0 \end{cases} \tag{20}$$

The learning rate factor in Equation 16 is a multiplication of a constant η set to a small positive value, $\hat{r}(t)$, and the perturbation $s(t)$ in Equation 21.

$$s(t) = \frac{Q'(t) - Q(t)}{exp(-\hat{r}(t-1))} \tag{21}$$

The multiplication $s(t)\hat{r}(t)$ means if a large perturbation leads to a great decision, then weights take an extra reward. If a large perturbation is not favorable, then it has minimal effect on weights values. Since ρ_V controls the meaning of MFs, the fixed rules become consistent with the change of weight values until VHIC achieves good performance.

5. PERFORMANCE EVALUATION

5.1 Simulation environment

We use simulator the 'ns-3' and the 'FuzzyLite' library to implement VHIC. The considered macro cell indoor scenario collocates Long Term Evolution (LTE) eNodeB with small access points WiFi1 and WiFi2. It involves UEs with low mobility ($\leq 3km/h$) and Voice over Internet Protocol (VoIP) traffic to evaluate the performance of VHIC. Initially, the monitored UE connects to WiFi1. Then while it moves, it detects WiFi2 and LTE connections and selects the appropriate access network according to VHIC. Table 1 summarizes the simulation parameters.

5.2 Implementation

In this paper, input variables are throughput and end to end delay. We assume that failure happens when the end to end delay is greater than 150ms [17]. EN has 15 weights initialized randomly to values in [-0.1,0.1]. The learning rate is $\beta = 0.3$, and the discount factor is $\gamma = 0.9$.

Figures 4 and 5 show the antecedent and consequent labels, respectively. Table 2 shows their initial definitions. Three labels describe the input variables: 'Low', 'Medium', and 'High'. The output variable is described by 9 labels: 'Very Low' (VL), 'Low' (L), 'Somewhat Low' (SL), 'Medium Low' (ML), 'Medium' (M), 'Medium High' (MH), 'Somewhat High' (SH), 'High' (H), and 'Very High' (VH). DMN uses nine fuzzy control rules, as shown in Table 3, that are constructed by using labels in Table 2. There are six antecedent labels in Layer 2, nine rules in Layer 3, and nine consequents in Layer 4. Softmin parameter is $\kappa = 10$. Learning rate is $\eta = 0.001$.

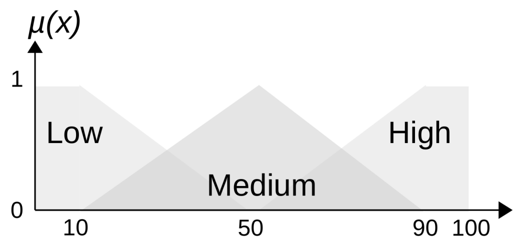

Fig.4: *Antecedents (throughput and end to end delay) labels*

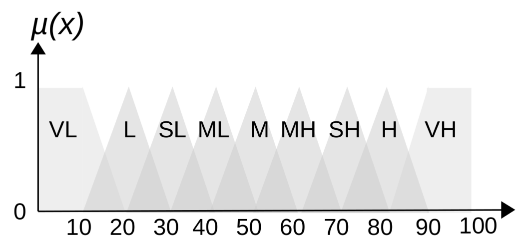

Fig.5: *Consequent (decision Q) labels*

5.3 Results

It would be interesting to compare the VHIC (learning FL) to VHDM-FL (VH Decision Making based on FL only). The latter is a FL approach similar to the fuzzy controller of VHIC [18]. The adopted performance metrics are throughput, end to end delay, jitter, Packet Loss Rate (PLR), decision delay, and a ratio of the decision to overall VH delays.

5.3.1 Throughput

Figure 6 represents the throughput obtained for VHDM-FL and VHIC. VHDM-FL throughput is stable at $20kbps$. On the other hand, VHIC enhances the throughput.

5.3.2 End to end delay

The end to end delay metric is relevant to analyze for real-time applications like VoIP. Figure 7 shows the performance results of end to end delay. For VHDM-FL, the end to end delay decreases from $0.5ms$ to $0.2ms$. In a brief time, VHIC lowers the end to end delay from $3.5ms$ to $0.1ms$. The VHIC failure signal depends on the end to end delay, and as learning proceeds, VHIC tries to eliminate failures. Consequently, VHIC reduces the end to end delay more quickly than VHDM-FL.

Table 1: *Simulation parameters*

Parameter	Value
UEs mobility model	SteadyStateRandomWaypointMobilityModel
Propagation loss model	indoorLossModel
Maximum queue delay for WiFi packets	500ms
Maximum queue size for WiFi	400 packets
WiFi Tx/Rx antenna gain	5dB
AP Tx power	18dBm
AP noise	5dBm
WiFi Rx Antenna	2
WiFi Tx Antenna	1
WiFi channel width	20MHz
UE noise	9dBm
UE Tx power	18dBm
UE Tx/Rx gain	0dBm
eNB TX power	46dBm
Macro eNB DL EARFCN	100
Macro eNB UL EARFCN	18100
Macro eNB bandwidth	25 RB
SRS periodicity	80
LTE RlcAm ReportBufferStatusTimer	20ms
LTE Tx buffer maximum size	10240B
LTE scheduler	RrFfMacScheduler
LTE spectrum channel type	MultiModelSpectrumChannel
eNB Antenna Model Type	ParabolicAntennaModel
eNB Antenna Beamwidth	70 degrees
eNB Antenna maximum attenuation	20dBm
EPS bearer	NGBR_VIDEO_TCP_DEFAULT
VoIP data rate	24kbps
VoIP packet size	60B
VoIP On/Off time	5s
VoIP traffic model	G.729A
VoIP packet inter-arrival time	20ms
ns-3 seed RngSeed	3
ns-3 number of trials RngRun	5

Table 2: *MFs parameters for antecedents (3 labels) and consequents (9 labels)*

Label	a	m	b	c	d	Label	a	m	b	c	d
Low	0	-	0	10	50	Very Low (VL)	0	-	0	10	20
Medium	10	50	90	-	-	Low (L)	10	20	30	-	-
High	50	-	90	100	100	Somewhat Low (SL)	20	30	40	-	-
						Medium Low (ML)	30	40	50	-	-
						Medium (M)	40	50	60	-	-
						Medium High (MH)	50	60	70	-	-
						Somewhat High (SH)	60	70	80	-	-
						High (H)	70	80	90	-	-
						Very High (VH)	80	-	90	100	100

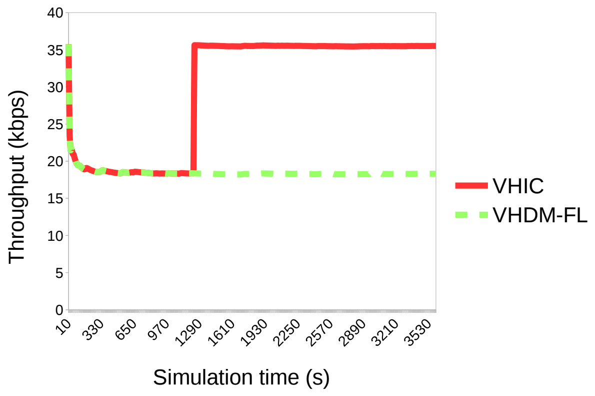

Fig.6: *Throughput*

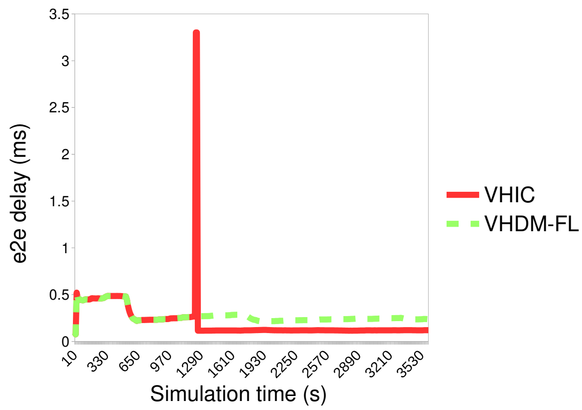

Fig.7: *End to end delay*

5.3.3 Jitter

Jitter is a relevant metric to analyze because high jitter values can lead to inadequate voice quality. Jitter must be $\leq 20ms$ to have good voice quality [17]. According to Figure 8, on average, VHDM-FL jitter is around $5\mu s$. For VHIC, jitter decreases from the average of $5\mu s$ to $0\mu s$. When the UE connects to WiFi, VHIC and VHDM-FL have some jitter. WiFi can be particularly bad for creating jitter, and the packet conflicts on WiFi increase with the increase in the number of devices operating on the same channel. As much as the learning proceeds, VHIC finds it reasonable to select LTE as the access network for the UE. That minimizes the jitter and keeps it near $0\mu s$.

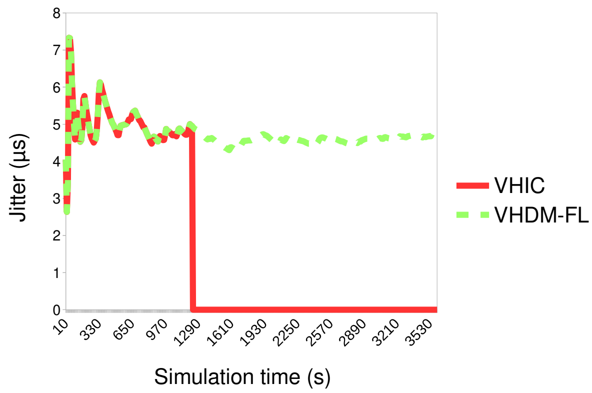

Fig.8: *Average jitter*

Table 3: *Decision control rules*

Throughput	End to end delay	Q
Low	Low	SH
Low	Medium	ML
Low	High	VL
Medium	Low	H
Medium	Medium	M
Medium	High	L
High	Low	VH
High	Medium	MH
High	High	SL

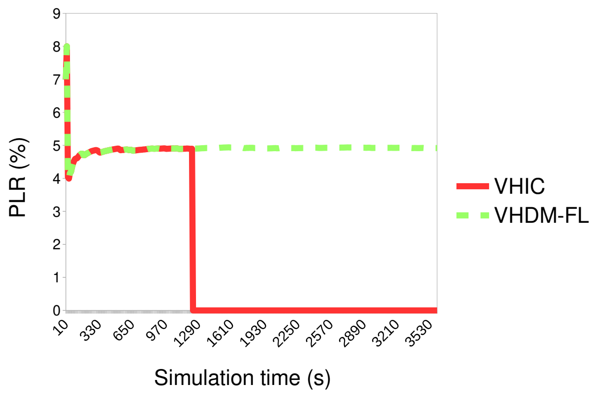

Fig.9: *Packet loss rate*

5.3.4 Packet loss rate

Figure 10 represents the PLR obtained for VHDM-FL and VHIC. On average, VHDM-FL PLR is 5%. On the other hand, the PLR of VHIC decreases to 0%.

5.3.5 Decision delay

Figure 9 represents the decision delay, which is the time duration to make a handover decision. VHDM-FL is faster than VHIC. The latter has a 92% higher decision delay than VHDM-FL because it uses learning capabilities. However, the VHIC decision delay does not exceed $1ms$, so it guarantees the low decision delay requirement for real-time applications. This result is valuable because nowadays, HetNets, mobile devices, and real-time applications require a pleasant performance level at a lower cost in terms of processing time.

5.3.6 Decision to overall VH delays

The decision to overall VH delays ratio is the ratio between decision and total VH delays (time duration between handover initiation and execution). Table 4

Table 4: *Decision delay to overall VH delay ratio*

Approach	Simulation time (s)	VH event	Overall VH delay (μ s)	ratio (%)
VHDM-FL	10	WiFi1 $\mapsto$ WiFi2	153	28
VHIC	10	WiFi1 $\mapsto$ WiFi2	674	84
	1240	WiFi2 $\mapsto$ LTE	668	84

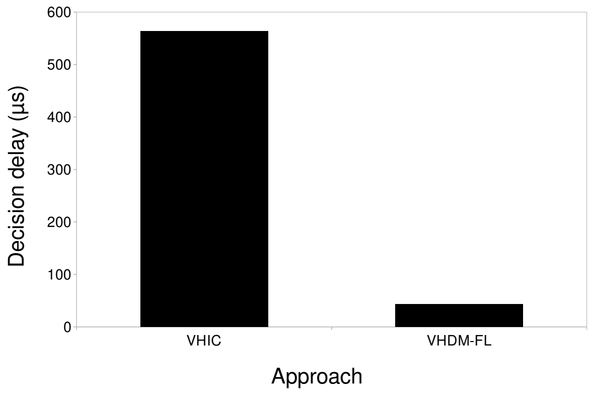

Fig.10: *Vertical handover decision phase delay*

shows the results of this ratio and VH events. The VH delay should be no more than a few hundreds of milliseconds. VHDM-FL and VHIC overall VH delays do not exceed one millisecond. Therefore, both approaches are fast and meet that condition to execute a seamless VH. VHDM-FL decision delay does not exceed 28% of the overall VH delay. VHIC decision takes 84% of the overall VH delay. Still, VHIC leaves enough time to the UE to properly switch to the selected CN and avoid handover failures.

Compared to VHDM-FL, VHIC consumes more time because of the learning part. If the knowledge is incomplete, wrong, or contradictory, then EN tunes DMN. In contrast, VHDM-FL relies on linguistic rules only instead of learning and has no formal approach for tuning. As we can see from the results, combining ANNs and FL in VHIC should unite their advantages and exclude their disadvantages. Due to its learning capabilities, VHIC can meet the requirements set for real-time applications better than VHDM-FL and makes a better trade-off between complexity, performance, and reliability.

6. CONCLUSION

This paper has highlighted the VH issue, which is a constant process in today's HetNets and usage scenarios. It has looked into ANN and FL based VH. Then, it has proposed an architecture named VHIC and based on FL, ANN, and RL. VHIC makes handover decisions according to application requirements and experience gained from past handover decisions. VHIC provides not only flexibility for initial deployment but also the adaptive capability to optimize the efficiency of VH with minimal human interference. Compared to the classical FL based VH, the learning included in VHIC yields better overall performance and reliability with a decision delay that respects the real-time requirement.

The next step in this research work will be to consider VHIC in medium and high mobility environments and realize it in some 5G networking architecture implementations. The goal is to see how results may change, and evaluate the scalability, complexity, performance, reliability, and real-time requirements support in such environment.

References

[1] T. E. Bogale and L. B. Le, “Massive MIMO and mmWave for 5G Wireless HetNet: Potential Benefits and Challenges,” in *IEEE Vehicular Technology Magazine*, vol. 11, no. 1, pp. 64-75, Mar. 2016.

[2] A. Ben Zineb, M. Ayadi and S. Tabbane, “ An Enhanced Vertical Handover Based on Fuzzy Inference MADM Approach for Heterogeneous Networks,” *Arabian Journal for Science and Engineering* , vol. 42 Issue 8, pp. 3263-3274, Aug. 2017.

[3] X. Li, F. Liu, Z. Feng, G. Xu and Z. Fu, “ A Novel Optimized Vertical Handover Framework for Seamless Networking Integration in Cyber-Enabled Systems,” *Future Generation Computer Systems*, vol. 79, pp. 417-430, Feb. 2017.

[4] S. Liang, Y. Zhang, B. Fan and H. Tian, “Multi-Attribute Vertical Handover Decision-Making Algorithm in a Hybrid VLC-Femto System,” in *IEEE Communications Letters*, vol. 21, no. 7, pp. 1521-1524, Jul. 2017.

[5] M. Subramani and V. Babu Kumaravelu, “ A Quality-Aware Fuzzy-Logic-Based Vertical Handover Decision Algorithm for Device-to Device Communication,” *Arabian Journal for Science and Engineering*, vol. 44, pp. 2413-2425, 2019.

[6] A. Sarma, S. Chakraborty and S. Nandi, “Deciding Handover Points Based on Context-Aware Load Balancing in a WiFi-WiMAX Heterogeneous Network Environment,” in *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 348-357, Jan. 2016.

[7] Y. Li, E. Datta, J. Ding, N. Shroff and X. Liu, “ Bandit Policies for Reliable Cellular Network Handovers in Extreme Mobility,” *arXiv:2010.15237v1*, 2020.

[8] D. E. Asuquo and F. E. Onuodu, “ A fuzzy ahp model for selection of university academic staff,”

International Journal of Computer Applications, vol. 975, pp. 88-87, 2016.

[9] C. W. De Silva, *Intelligent control: fuzzy logic applications*, CRC press, 2018.

[10] Y. Bi, L. Tian, M. Liu, Z. Liu and W. Chen, "Research on Joint Handoff Algorithm in Vehicles Networks," *Chinese Journal of Engineering*, vol. 2016, Article ID 3190264, 10 pages, 2016.

[11] M. Khan, A. Ahmad, S. Khalid, S. Hassan Ahmed, S. Jabbar and J. Ahmad, " Fuzzy based multi-criteria vertical handover decision modeling in heterogeneous wireless networks," *Multimedia Tools and Applications*, vol. 76, no. 23, pp. 24649-24674, 2017.

[12] B. Naeem, R. Ngah and S. Z. Mohd Hashim, " Reduction in ping-pong effect in heterogeneous networks using fuzzy logic," *Soft Computing*, vol. 23, no. 1, pp. 269-283, 2019.

[13] A. Gómez-Andrades, P. Muñoz, I. Serrano and R. Barco, " Automatic root cause analysis for LTE networks based on unsupervised techniques," in *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2369-2386, Apr. 2016.

[14] Z. Wang, L. Li, Y. Xu, H. Tian and S. Cui, "Handover Control in Wireless Systems via Asynchronous Multiuser Deep Reinforcement Learning," in *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4296-4307, Dec.2018.

[15] Z. Ali, N. Baldo, J. Mangues-Bafalluy and L. Giupponi, "Machine learning based handover management for improved QoE in LTE," in *Proc. NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, pp. 794-798, 2016.

[16] H. R. Berenji and P. Khedkar, " Learning and tuning fuzzy logic controllers through reinforcements," *IEEE Transactions on neural networks*, vol. 3, no. 5, pp. 724-740, Sep. 1992.

[17] M. H. Miraz, S. A. Molvi, M. Ali, M. A. Ganie and A. H. Hussein, "Analysis of QoS of VoIP traffic through WiFi-UMTS networks," *arXiv preprint arXiv:1708.05068*, 2017.

[18] S. Driouache, N. Naja, and A. Jamali, "Fuzzy Logic Based Intelligent Vertical Handover Decision in Heterogeneous Networks," in *International Conference on Artificial Intelligence & Industrial Applications*, pp. 99-109, 2020.

Saida Driouache is currently a doctoral student in Computer Science working under the supervision of Professors Najib Naja, and Abdellah Jamali, at Institut National des Postes et Télécommunications, Morocco. She obtained her engineering degree in Networks and Systems in 2013, from Ecole Nationale des Sciences Appliquées, Oujda, Morocco. Her research is focused on Communication Networks, Mobility Management, Performance Evaluation, and AI.

Najib Naja Currently, He is a professor in Institut National des Postes et Télécommunications, Morocco since 1994 and a permanent member of research Laboratory STRS, 'Networks, Architectures, Informatics and Systems Security' Team since 2012. He received his engineering degree in Computer Science in 1989, from Ecole Nationale Superieure des Télécommunications, Bretagne, Brest, France). He obtained his Ph.D. degree in Telecommunications from Rennes University, France, in 1994. He had the state Doctorate degree of Computer Science from Mohammed 5 University, Morocco, in 1997. His research interests include the Distributed Computing, Wireless Networks, Signal Processing, QoS in Wireless Networks, Network Performance Analysis.

Abdellah Jamali is a professor at Ecole Nationale des Sciences Appliquées, Settat, Morocco, and permanent member of research Laboratory IR2M at Hassan 1 University, Morocco, since October 2011. He received a thesis in Computer Science from Hassan 2 University, Morocco. He is a founding member since 2007, of a research group: e-Next Generation Networks (e-NGN) for Africa and Middle East. His research interests include Computer Networks, QoS, Networks Performance Analysis.