

# Health-Flow: Criticality-Aware Flow Control for SDN-Based Healthcare IoT

Sudip Misra<sup>1</sup>, Ruelia Saha<sup>2</sup>, and Nurzaman Ahmed<sup>3</sup>

<sup>1,2,3</sup>Department of Computer Science & Engineering  
Indian Institute of Technology  
Kharagpur, India-721302

<sup>1</sup>sudipm@iitkgp.ac.in, <sup>2</sup>rueliasahacse@iitkgp.ac.in, <sup>3</sup>nurzaman@cse.iitkgp.ac.in

**Abstract**—In this paper, we propose a criticality-aware traffic forwarding scheme for mobile devices to maximize the efficiency of the software-defined healthcare network. We implement machine learning based-approach to find the criticality of flows combining the location of a person in the system. Concerning the levels of the criticality in traffic, the proposed protocol dynamically places or removes flow-rules at the edge access points. Consequently, helps in taking adequate actions for the incoming requests adaptively with better network reconfiguration overhead, latency, and energy consumption. We mathematically formulate the resource reallocation problem in terms of optimization by minimizing the network overhead subject to criticality requirements of the packet flows. The proposed scheme has the potential to reduce latency by 52%, overhead by 19%, and energy consumption by 12% as compared to the existing schemes.

**Index Terms**—Software-Defined Network (SDN), Internet of Things (IoT), Criticality, Smart Healthcare

## I. INTRODUCTION

In the recent COVID-19 pandemic situation, several challenges are associated with the healthcare ecosystem [1], [2]. Specifically, the underdeveloped countries are facing a lot of issues related to insufficient healthcare infrastructure. Due to a lack of adequate accommodation in the hospitals, the condition of the patients is not getting diagnosed on time, thereby leading to an unnecessary deterioration of the patient's condition and deaths. For instance, one of the solution is the use of Internet of Things (IoT) and mobile devices such as fitness trackers, smartphones, and sensors/actuators. Thus, it is essential to accommodate more heterogeneous and mobile devices, which is often difficult to handle by the traditional network. The advancement of Software-Defined Internet of Things (SD-IoT) makes it possible to efficiently integrate multiple heterogeneous healthcare-related sensors and devices to the Internet [3], [4]. Decoupling the controlplane from the dataplane in Software-Defined Network (SDN) gives the feasibility of configuring the controller depending on the requirement of the users and henceforth avoiding the vendor-specific architecture present in the physical devices. Dataplane consists of SDN switches that route the packets, and the controllers that use OpenFlow protocol [5] to control the dataplane devices.

The current generation is undergoing stress and lack of required time for maintaining physical fitness, thereby resulting in developing the physiological disorder at an early age. With the emergence of fitness trackers in the recent past, it is possible

to keep track of the physiological parameters continuously. The different sensors that are present in the monitoring devices include pulse rate, SpO2, calorimeter, sleep tracker, and location tracker sensor. Due to the criticality, the traffic generated by these sensors needs some distinctive forwarding in the SDN. Wireless access networks such as Wireless Local Area Networks (WLANs) and Wireless Personal Area Networks (WPANs) are the most promising solutions to connect a massive number of mobile IoT devices. Software-Defined Access Network (SDAN) [6] combines such networks with SDN for better control in terms of traffic forwarding, channel allocation, etc. An Access Point (AP) works as SDN data-plane, holds the critical responsibility to connect them with the Internet. Besides fitness trackers, a smartphone in people's pockets uses sensors and radio interfaces to monitor and communicate, respectively. Fig. 1 shows a healthcare network architecture having various devices (mobile/static) connected to the Internet through AP/Edge devices. To deal with the critical health-related traffic flows, the current SDN does not have any mechanism. The Quality of Service (QoS) demands of such applications to be guaranteed during the communication between the APs and cloud service providers.

APs are resource constraints in terms of storage and energy consumption. With the increase in the mobility and heterogeneity in the network, the flow table present within the APs comprising the data plane also increases. Thus, managing them systematically and maintaining a fixed storage space is often difficult. The existing solutions deal with the proactive flow-rule replacement policy. Moreover, the flow-rules parameters comprise of the significant packet header fields, thereby incurring more memory space. The architecture addresses these lacunae and proposes a novel approach to fulfilling these gaps. In this paper, we present a dynamic flow control mechanism for high priority traffic, named, *Health-Flow* for mobile nodes. The key contributions of this paper are given below:

- We propose a criticality-aware efficient flow-rule placement scheme for smart healthcare by reading the physiological parameter value.
- An ML-based SDN controller, called *Health Controller (HC)*, is developed to generate a health-related notification and predict the criticality index and location of the mobile user.

## II. RELATED WORK

Researchers have proposed several flow-rule replacement schemes for improving the efficiency of SDN. Danielis *et al.* [7] considers the problem of dynamic migration of flow-rule and proposed an algorithm for finding a feasible flow migration solution directly in polynomial time complexity. The author also put forward an algorithm for computing flow migration, both directly and indirectly. Bera *et al.* [8] proposes an adaptive flow-rule placement policy. It provides statistics regarding per-flow to the SDN controller, thereby enhancing overall network performance. The author proposed a scheme for redistributing rules on encountering rule congestion at a switch and accommodating new flows in the network. Vissicchio *et al.* [9] preserves forwarding policy while updating the SDN network. It combines the dual feature that exists between the addition and replacement of flow table rules in the switch. It recognizes restrictions on rule additions and replacements that are required for preventing the occurrence of policy violations during the update. Li *et al.* [10] proposes an effective rule management strategy for optimizing the rule replacement, specifically for mobile users. Bera *et al.* [11] proposes a flow-rule placement algorithm, called *MobiFlow*, for mobile nodes present in the network. The author minimizes the cost related to the flow-rule placement by formulating integer linear programming. As solving the ILP is NP-hard, the author further proposed a greedy method to obtain the optimal number of APs required. Saha *et al.* [12] describes two types of QoS routing strategies, which include delay-sensitive- to handle delay-sensitive flows and loss-sensitive: to deals with loss-sensitive flows, thereby maximizing the overall network performance. The author also proposed Yen's K-shortest paths algorithm based greedy method for computing the optimal path for forwarding. Table I compared the present works with the proposed solution.

TABLE I: Comparison of Health-Flow with existing solutions for enabling efficient flow management

Protocol	QoS	Hybrid	Mobility	Criticality	Trajectory
[7]	✓	✗	✓	✗	✓
[9]	✓	✗	✗	✗	✗
[11]	✓	✗	✗	✗	✓
[10]	✓	✗	✓	✗	✗
[12]	✓	✗	✗	✗	✓
<b>Proposed work</b>	✓	✓	✓	✓	✓

The already stated literature survey reveals that the existing flow-rule replacement policies are proactive, and there exists lacuna in hybrid flow-rule replacement policy (i.e., which includes both proactive and reactive flow-rule placement scheme). Proactive flow-rule placement scheme involves populating the flow table with all possible predefined flow-rules. On the other hand, reactive flow-rule placement scheme involves designing of flow-rules on the fly on encountering a new packet and storing it in the flow table. The existing solutions are not very suitable for the mobile healthcare network. In the proposed

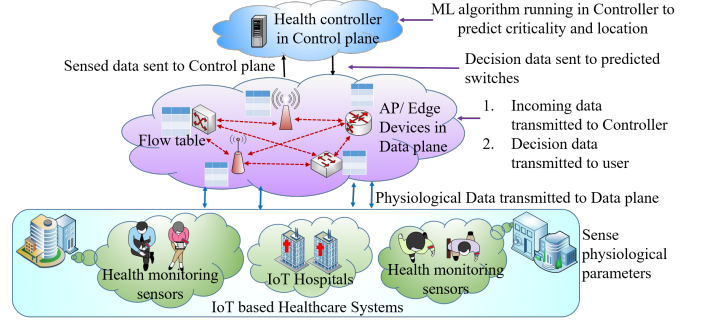


Fig. 1: SDAN for Smart Healthcare

architecture, we establish a hybrid flow-rule placement policy, along with the introduction of the criticality index value of every packet.

## III. THE PROPOSED PROTOCOL

We propose a criticality-aware dynamic flow-rule placement scheme for the healthcare network.

### A. System Model

The overview of the proposed SDAN network architecture is shown in Fig. 1.

### B. Criticality and Location Aware Flow Control

Let us consider that a set of fitness trackers  $F = \{F_1, F_2, \dots, F_n\}$  are present in the network and  $S = \{S_1, S_2, \dots, S_k\}$  be the set of sensors in every fitness trackers. The sensed physiological parameters read by each sensor node  $j$  present in each fitness tracker  $i$  is represented by  $P_{ij}$ . After acquiring the physiological parameter values from the sensor nodes, the data are transmitted to the nearest AP. Let us consider that a set of access points  $AP = \{AP_1, AP_2, \dots, AP_m\}$  are present in the proposed network architecture. After receiving the sensed data, AP transmits the data to the health controller. Health controller has pre-installed Machine Learning (ML) code in it. The ML algorithm processes the received data to compute and predict the criticality of the person. If the criticality of a particular person  $C_{ij}$  is beyond a predefined threshold value  $C_{thres}$ , then it signifies that the condition of the person is critical, thereby the person needs to be notified about the required steps to be taken at the earliest possibility. In such a situation, if the person is mobile, then the expected trajectory path of the person is predicted by the Gaussian Mixture Model, and APs closest to the trajectory are selected. The flow-rule for notifying the person to take the required steps is then stored in the identified APs (i.e., proactive placement of flow-rules in the flow table). As the person is mobile, the person may move away from the forwarding devices and may be close to some other forwarding device. Thus, updating the flow table present within the nearest AP is necessary rather than updating it in the previous AP that transmitted the data. If the condition of the patient is not beyond a specified threshold, then further notification and identification of the paths of the person are omitted. In the future, such rules are generated on

the fly (i.e., flow-rules are generated reactively depending on the requirement).

Markov Chain-based [13], [14] and Conditional Random Fields-based [15] location prediction schemes use the current location for finding the next location. As people's trajectories mostly have multiple mobility patterns due to subjective destination, objective environment, and people's movement. Therefore, such trajectory patterns can be represented by a Gaussian process and the entire trajectories thus correspond to a Gaussian Mixture Model (GMM) [16], [17]. The history of the trajectory can be written as:

$$\begin{aligned}\vec{G} &= \{(\vec{x}_1, \vec{y}_1), (\vec{x}_2, \vec{y}_2), (\vec{x}_3, \vec{y}_3), \dots, (\vec{x}_n, \vec{y}_n)\} \\ &= (\vec{X}, \vec{Y})\end{aligned}\quad (1)$$

where  $\vec{T}_i = (\vec{x}_i, \vec{y}_i)$  is the trajectory of  $i$ th device and  $(\vec{X}, \vec{Y})$  is the mapping vector of the trajectories in X and Y directions. For training set  $\vec{G}$ , likelihood function can be calculated as:

$$\begin{aligned}P(\vec{X}|\gamma) &= \prod_{n=1}^{N_p} p(\vec{x}_n|\gamma) \\ P(\vec{Y}|\gamma) &= \prod_{n=1}^{N_p} p(\vec{y}_n|\gamma)\end{aligned}\quad (2)$$

where  $p(\vec{x}_n|\gamma)$  and  $p(\vec{y}_n|\gamma)$  represents the probability function of trajectory's X and Y direction, respectively.  $N_p$  is the number of trajectory's patterns and  $\gamma = \{\alpha_i, \beta_i, \delta_i\}; i = 1, 2, 3, \dots, N_p$ , where,  $\alpha_i$  is the weight,  $\beta_i$  is the means and  $\delta_i$  is the co-variance of  $i^{th}$  trajectory patterns. The process of forecasting involves mainly three steps: (1) application of Gaussian Mixture clustering method to trajectory dataset  $G$  to obtain  $M$  clusters, which correspond to  $M$  different trajectory patterns; (2) estimation of parameters by implementing expectation-maximization algorithm; (3) finally predicting the future location of a mobile user based on his/her recent trajectory. We use Stanford Drone Dataset [18] to study the trajectory patterns including outdoor movements, and activity routines. The likelihood of location is used in the proposed flow-rule placement scheme.

The memory space available within the edge devices is limited. If the criticality index of any packet is greater than the specified threshold, the controller generates packet containing information related to the immediate action to be taken and the corresponding flow-rule is stored in the predicted APs. Thus, for the time-critical data, the newly generated flow-rule of the corresponding packet is updated in the nearby edge devices. Hence, in the proposed architecture, the memory space is conserved by eliminating packet generation by the controller and storing flow-rule within the forwarding device, if the criticality is below a specified threshold value for a particular packet. The threshold value is predefined and it is used in deciding the criticality.

In the proposed architecture, it is evident that we consider not only up-link transmission of the sensed data from person to the health controller but also after processing the data, down-link transmission of the notification about the abnormality, and

---

**Algorithm 1:** Criticality-Aware Flow Control for Mobile Devices in Smart Healthcare

---

**Inputs:** Set of user,  $P = P_1, P_2, P_3, \dots, P_q$

**Output:** Criticality aware packet forwarding

---

```

1 Sense the physiological parameter data of Q and
  transmit them to switches
2 for  $i = 1$  to  $q$  do
3   Switch transmits the packet of  $Q_i$  to the health
    controller (HC)
4   HC apply ML algorithm to predict criticality index
     $ci_i$  of  $Q_i$ 
5   if  $(ci_i \geq c_{thres})$  then
6     HC generates packet containing health
      notification information
7     HC executes GMM based location prediction
      schemes for predicting the expected location
8     HC updates the flow table by the flow-rule in
      the identified APs
9     Downlink transmission of notification packet
      from switch to  $Q_i$ 
10  else
11    No notification sent to user

```

---

the required action to be taken by the person from the health controller.

### C. Delay incurred

The physiological parameter data after getting sensed by the fitness trackers are sent to the nearest AP. After calculating the criticality, if criticality index value is beyond a specified threshold value, it gets transmitted to the predicted APs, which are most adjacent to the person for sending a notification. Thus, the total delay incurred in performing these tasks is given below:

$$T_{tot} = \begin{cases} T_{upl} + T_{ci}, & \text{if } ci < ci_{thres} \\ T_{upl} + T_{ci} + T_{path} + T_{dwl}, & \text{if } ci > ci_{thres} \end{cases} \quad (3)$$

where  $t_{upl}$  is the time taken to transmit the sensed physiological data to the nearest AP.  $T_{ci}$  denotes the time taken by the ML algorithm to calculate the criticality index of the person. Delay incurred in predicting the trajectory of the person, depending on the previous record and updating the flow table in APs nearest to those trajectories, is represented by  $T_{path}$ .  $T_{dwl}$  is the total time needed to transmit the notification to the person from the AP,  $ci$  is the criticality index of the person, and  $ci_{thres}$  represents the threshold criticality value.

### D. Energy consumed

The total energy consumed by the proposed architecture is as follows:

$$E_{tot} = \begin{cases} E_{upl}, & \text{if } ci < ci_{thres} \\ E_{upl} + E_{AP} + E_{dwl}, & \text{if } ci > ci_{thres} \end{cases} \quad (4)$$

where  $E_{upl}$  is the energy consumed by the fitness tracker to transmit the sensed physiological data to the nearest AP, energy consumed in updating the flow table in those APs closest to the predicted trajectories is represented by  $E_{AP}$ .  $E_{dwl}$  represents the total energy consumed by the active AP to transmit the notification to the person from the AP.  $ci$  represents the criticality index of the person, and  $ci_{thres}$  represents the threshold criticality value.

### E. An Illustrative Example

Let us consider a scenario where several mobile devices are sensing physiological data of the respective persons carrying those devices. These devices include mobile phones, tablets, and fitness trackers, having fitness tracking software pre-installed. These devices continuously sense data and transmit it to the nearby APs. As all the persons present in the network, change their locations, uplink and downlink data transmission may not be from the same AP. Due to the limited storage space at AP, it is convenient to update the flow table with the newly designed flow-rules only within the APs closest to the person to whom data need to be transmitted. Interchange in the location among the devices and the necessary updation in the required flow table is shown in Fig. 2.

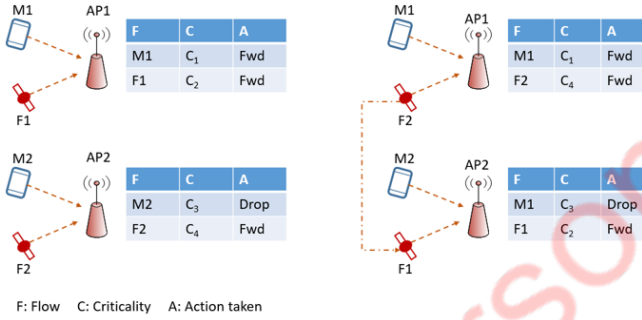


Fig. 2: Interchange in the location of the requesting devices and the updation of the flow table in the expected APs on encountering motion of the person

If the next location of the person is unpredictable, it is highly likely that the person is present in one of the  $k$  location,  $L = L_1, L_2, \dots, L_k$ . If the criticality of a person is beyond the specific threshold value, then all the possible APs nearest to those locations are updated with the newly designed flow-rules (as shown in Fig. 3).

### F. Effective Flow-rule Placement

We develop an effective flow-rule placement scheme considering both reactive and proactive approach. Figs. 2 and 3 represent the dynamic change in flow-rules at APs with the change in the position of a person with a critical physiological condition. The algorithm for criticality-aware flow control is presented in Algorithm 1. Let us consider a set of APs,  $AP = \{AP_1, AP_2, \dots, AP_m\}$  are present in the proposed network architecture and  $P = \{P_1, P_2, \dots, P_q\}$  persons present in the network. After receiving the sensed data from the fitness tracker,

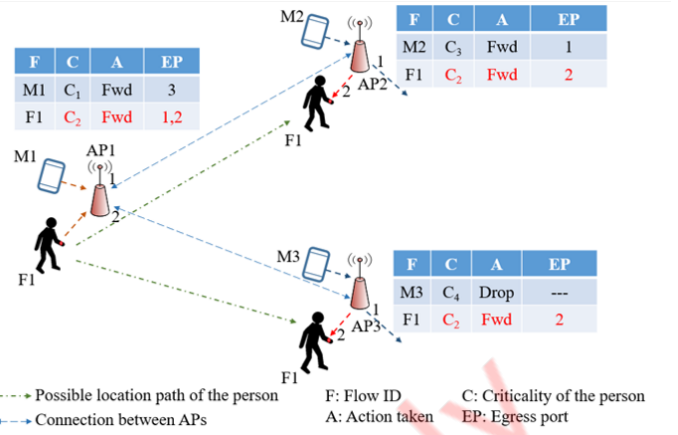


Fig. 3: Identifying the expected location of the person by the trajectory and updating the predicted APs' flow table nearest to the person.

it is processed by the controller to compute and predict the criticality of the person as shown in steps 1-4. If the criticality of a particular person  $C_{ij}$  is beyond a predefined threshold value  $C_{thres}$ , it signifies that the condition of the person is critical, thereby the person needs to be notified about the required steps to be taken at the earliest possibility, as mentioned in step 5-6. In such a situation, if the person is mobile, then the expected trajectory path of the person is predicted along with the set of APs present in that trajectories, as given in step 7. The flow-rule for notifying the person to take the required steps is then stored in the identified APs (i.e., proactive placement of the flow-rule in the flow table), as mentioned in step 8 and is finally transmitted to the person as stated in step 9. Since the APs present in the network are resource constraints, the flow-rules for critical data is stored, excluding the flow-rule for the other data as mentioned in step 10-11.

Let us consider that the maximum number of flow-rules that can be placed in  $i^{th}$  AP considering the capacity constraints nature of AP is  $R_{max}^i$  [14], which can be calculated as,, where  $A_i(R_j)$  represents the  $j^{th}$  flow-rule in  $i^{th}$  AP, and  $n$  is the total number of flow-rules currently available at the AP. Therefore,  $R_j$  correlates to the  $j^{th}$  flow-rule in an AP. SDN controller aims to minimize the number of activated APs so that the overall cost in the network is minimized. Mathematically,

$$\text{Minimize } \sum_{i \in A} A_{act,i}^t \quad (5)$$

subject to

$$R_i^t \leq R_i^{max}, i \in A \quad (6a)$$

$$r_j^t \geq 1, j \in U \quad (6b)$$

$$r_j^t \in R_i^t = TRUE, \text{ if } j - > i \text{ is } TRUE \quad (6c)$$

$$C_j \geq C_{th} \quad (6d)$$

$$E_i^t \leq E_i^{avl}, i \in A \quad (6e)$$

$$D_j^t \leq D_j^{th} \quad (6f)$$



where  $A_{act,i}^t$  signifies that an AP,  $i \in A$ , is activated at a particular time instance  $t$ . Equation 6a represents that the total number of flow-rules available at the  $AP_i$  at time instance  $t$ , do not exceed the maximum number of flow-rule capacity. Equation 6b signifies that the number of flow-rules related to a user must not be less than '1' as a flow-rule related to a person may be present at multiple APs if the condition of the person is critical and prompt response need to be taken. It is worthy of being noted that the flow-rule for a specific packet must be placed into the appropriate flow table of the AP, which is associated with the person. This is introduced by the constraint, as already indicated by Equation 6c. Equation 6d denotes that the criticality index of the person is above the specified threshold, and flow-rules for those packets are considered for updation in the flow table associated with the person. Equation 6e denotes that the total energy required to satisfy all the requests must not exceed the available energy,  $E_{avl}$ , at the associated AP. Finally, the delay incurred to process a particular request coming from a person that must not be greater than the maximum delay is represented by Equation 6f. The above-stated optimization problem is mathematically formulated as an integer linear programming (ILP). It is to be noted that finding an optimal solution to the above-stated problem having polynomial time complexity is NP-hard.

TABLE II: Simulation Parameters

Parameter	Value
Simulation area	$500 \times 500 \text{ m}^2$
Number of edge nodes	50 – 250
Size of data set	100 – 300
Mobility model of APs	<i>ConstantPosition</i>
Edge technology	<i>WiFi</i>
User's deployment strategy	<i>Uniform – Random</i>

#### IV. PERFORMANCE EVALUATION

The performance of the proposed architecture is evaluated using different simulation parameters listed in the Table II. We consider both small scenarios as well as large scenarios by varying the number of users and APs present in the IoT scenario. Detailed analysis of Health-Flow protocol is compared with the traditional SDN, and Mobi-Flow protocol [14] are discussed below.

##### A. Delay incurred

One of the main objectives of Health-Flow is to reduce the total delay incurred in alerting the person about his/her abnormal physiological condition. Fig. 4 depicts that the total delay incurred in updating the flow table present in every switch reduces significantly. It is because Health-Flow first checks the criticality of the person. Thus if the condition of a person is not critical, then it saves time in updating the flow-rule in all the flow tables present in APs. Moreover, if the condition of the person is critical, then it uses the Gaussian mixture model to predict all possible locations of the person and updating only those flow tables present in the identified APs closest to

the predicted locations. Thus, the time in updating the flow table present in all the APs is significantly reduced. Therefore, the proposed scheme provides better results in lowering delay compared to the existing network architecture.

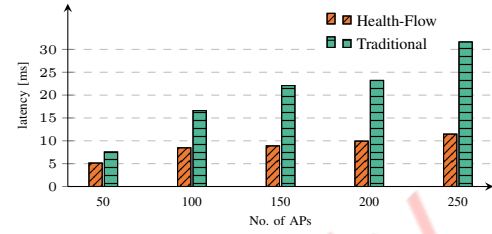


Fig. 4: Total delay incurred with increasing number of APs

##### B. Energy consumption

Fig. 5 shows the overall energy consumed by Health-Flow as well as the traditional network. The proposed architecture performs better in contrast to the conventional network. It is because of the Health-Flow architecture, as mentioned earlier, updates the flow table in only those APs, which may be close to the predicted position of the person. Furthermore, it updates the required flow tables only when the criticality of the person is beyond a specified threshold value; otherwise, the flow tables remain untouched. Thus, the scheme reduces the extra energy required in updating the flow tables present in every APs, thereby allowing other APs to stay in the inactive state and save energy. Furthermore, from Fig 5, it is evident that the proposed architecture consumes lesser energy compared to Mobi-Flow. It is because Health-Flow eliminates designing flow-rule for the packet whose criticality index value is smaller than a specified threshold value.

##### C. Overhead involved

We measure the control overhead in the proposed architecture and compared it with Mobi-Flow and traditional scheme. As shown in Fig. 6, control overhead in Health-Flow is lesser than the traditional architecture. This is caused by the absence of a mobility prediction method in the existing scheme. Moreover, the combination of proactive and reactive flow-rule placement allows reducing overall overhead in the network optimally by the dynamic flow-rule placement strategy.

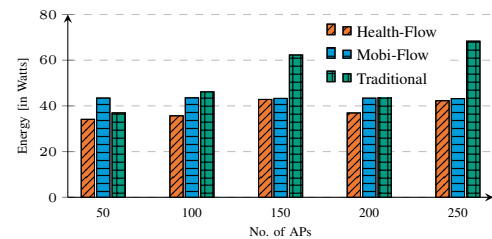


Fig. 5: Total energy consumed

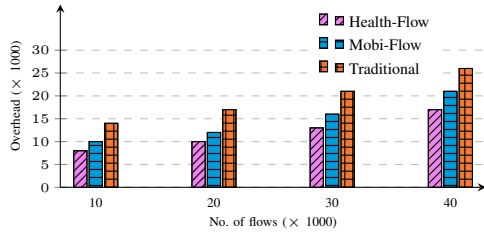


Fig. 6: Total control overhead

#### D. Accuracy of ML algorithm

One of the main objectives of Health-Flow is to predict the criticality of the user by analyzing the physiological parameters. Thus it is essential to identify the most suitable ML algorithm that fits best in the proposed scenario. We use Kaggle's Heart disease dataset[19] to study the physiological condition of a person and predict whether the person is suffering from heart disease by calculating the criticality index using ML algorithms. From Fig. 7, it can be observed that when the data set is small, all the algorithms have similar performance, but with the increase in the size of the dataset, the neural network has a better performance compared to the other two ML algorithms. Furthermore, we use GMM to cluster the different modes of transport of a person for determining the trajectory of the person.

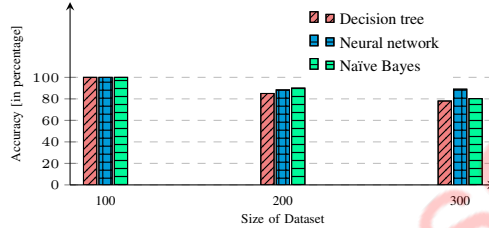


Fig. 7: Achieved accuracy of the applied ML methods

#### V. CONCLUSION

This paper introduced Health-Flow, a hybrid flow-rule replacement scheme that depended on the criticality and mobility of the users. The proposed protocol used ML algorithms to predict the criticality of a person by reading the physiological parameters, and based on the conditions, required notifications were sent to the person for immediate action. Applying GMM, we predicted all possible trajectories of the person from the existing dataset, and the nearest AP's flow tables were also updated with the newly designed flow-rule so that notifications can be sent to the person at the earliest convenience. By performing extensive simulation, it was observed that Health-Flow outperformed traditional networks in terms of energy consumption, the delay incurred, and control overhead. We considered that the entire network includes SDN, and thus there exists homogeneity in the system, but in reality, the whole network may not deploy the SDN architecture. Therefore, the future scope of work includes proposing architecture for different scenarios. Furthermore, we intend to implement the

proposed architecture in a real situation to have a better understanding of the feasibility.

#### ACKNOWLEDGMENT

The authors gratefully acknowledges the funding support received from INAE (Sanction letter no. INAE/121/AKF, Dt. 13-02-2019), and SERB/IMPRINT-II (Sanction letter no SERB/F/12680/2018-2019;IMP/2018/000451, Dt. 25-03-2019) for executing parts of this project.

#### REFERENCES

- [1] W. H. Organization *et al.*, "Coronavirus disease 2019 (COVID-19): situation report, 92," 2020.
- [2] J. J. Van Bavel, K. Baicker, P. S. Boggio, V. Capraro, A. Cichocka, M. Cikara, M. J. Crockett, A. J. Crum, K. M. Douglas, J. N. Druckman, *et al.*, "Using social and behavioural science to support COVID-19 pandemic response," *Nature Human Behaviour*, pp. 1–12, 2020.
- [3] P. Martinez-Julia, A. F. Skarmeta, *et al.*, "Empowering the Internet of Things with Software Defined Networking," *White Paper, IoT6-FP7 European research project*, 2014.
- [4] T. Ninikrishna, S. Sarkar, R. Tengshe, M. K. Jha, L. Sharma, V. Daliya, and S. K. Routray, "Software defined IoT: Issues and challenges," in *International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 723–726, IEEE, 2017.
- [5] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, *et al.*, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 87–95, 2014.
- [6] K. J. Kerpez, J. M. Cioffi, G. Ginis, M. Goldberg, S. Galli, and P. Silverman, "Software-defined access networks," *IEEE Communications magazine*, vol. 52, no. 9, pp. 152–159, 2014.
- [7] P. Danielis, G. Dán, J. Gross, and A. Berger, "Dynamic flow migration for delay constrained traffic in software-defined networks," 12 2017.
- [8] S. Bera, S. Misra, and A. Jamalipour, "FlowStat: Adaptive Flow-Rule Placement for Per-Flow Statistics in SDN," *IEEE Journal on Selected Areas in Communications*, vol. 37, pp. 530–539, March 2019.
- [9] S. Vissicchio and L. Cittadini, "Flip the (flow) table: Fast lightweight policy-preserving sdn updates," in *International Conference on Computer Communications*, April 2016.
- [10] H. Li, P. Li, and S. Guo, "Morule: Optimized rule placement for mobile users in sdn-enabled access networks," in *Global Communications Conference*, pp. 4953–4958, Dec 2014.
- [11] S. Bera, S. Misra, and M. S. Obaidat, "Mobi-Flow: Mobility-Aware Adaptive Flow-Rule Placement in Software-Defined Access Network," *IEEE Transactions on Mobile Computing*, vol. 18, pp. 1831–1842, Aug 2019.
- [12] N. Saha, S. Bera, and S. Misra, "Sway: Traffic-Aware QoS Routing in Software-Defined IoT," *IEEE Transactions on Emerging Topics in Computing*, vol. PP, pp. 1–1, 06 2018.
- [13] W. Mathew, R. Raposo, and B. Martins, "Predicting future locations with hidden Markov models," in *ACM conference on ubiquitous computing*, pp. 911–918, 2012.
- [14] S. Bera, S. Misra, and M. S. Obaidat, "Mobi-flow: Mobility-aware adaptive flow-rule placement in software-defined access network," *IEEE Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1831–1842, 2018.
- [15] R. Pan, J. Zhao, V. W. Zheng, J. J. Pan, D. Shen, S. J. Pan, and Q. Yang, "Domain-constrained semi-supervised mining of tracking models in sensor networks," in *International conference on Knowledge discovery and data mining (SIGKDD)*, pp. 1023–1027, 2007.
- [16] N. T. An and T. M. Phuong, "A gaussian mixture model for mobile location prediction," in *International Conference on Research, Innovation and Vision for the Future*, pp. 152–157, IEEE, 2007.
- [17] Q. Peng, M. Zhou, Q. He, Y. Xia, C. Wu, and S. Deng, "Multi-objective optimization for location prediction of mobile devices in sensor-based applications," *IEEE Access*, vol. 6, pp. 77123–77132, 2018.
- [18] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European conference on computer vision*, pp. 549–565, Springer, 2016.
- [19] "Heart Disease UCI." /<https://www.kaggle.com/ronitf/heart-disease-uci>. [Online].