Energy Efficient Contact Tracing and Social Interaction based Patient Prediction System for COVID-19 Pandemic

Charuka Moremada, *Student Member, IEEE,* Chamara Sandeepa, *Student Member, IEEE,* Nadeeka Dissanayaka, *Student Member, IEEE,* Tharindu Gamage, and Madhusanka Liyanage, *Senior Member, IEEE*

Abstract—Due to the spread of Coronavirus disease 2019 (COVID-19), the world has encountered an ongoing pandemic to date. It is a highly contagious disease. In addition to the vaccination, social distancing and isolation of patients are proven to be one of the commonly used strategies to reduce the spread of disease. For efficient social distancing, contact tracing is a critical requirement in the incubation period of 14-days of the disease to contain any further spread. However, we identify that there is a lack of reliable and practical social interaction tracking methods and prediction methods for the probability of getting the disease. This paper focuses on user tracking and predicting the infection probability based on these social interactions. We first developed an energy-efficient BLE (Bluetooth Low Energy) based social interaction tracking system to achieve this. Then, based on the collected data, we propose an algorithm to predict the possibility of getting the COVID-19. Finally, to show the practicality of our solution, we implemented a prototype with a mobile app and a web monitoring tool for healthcare authorities. In addition to that, to analyze the proposed algorithm's behaviour, we performed a simulation of the system using a graph-based model.

I. Introduction

A New emergence of an epidemic was identified in late December 2019. This is known to be first detected in Wuhan city, China. The virus causing the pandemic is named Severe Acute causes Respiratory Syndrome Coronavirus 2 (SARS-CoV-2), and the disease caused by this virus was called COVID-19 by the World Health Organization [1]. This epidemic has gradually turned into a global pandemic. Many countries faced great distress in their health systems where hospitals and medical staff capacity were insufficient to treat the rapid increase of COVID-19 patients. A disease like this can cause a tremendous life-threatening risk to these patients since the healthcare systems run out of their capacity to treat them properly. However, if the authorities can make early identification of potential COVID-19 patients via contact

Charuka Moremada, Chamara Sandeepa, Nadeeka Dissanayaka, Tharindu Gamage was with the Department of Electrical and Information Engineering, University of Ruhuna, Galle, Sri Lanka.
E-mail: charuka.kavinda@yahoo.com, cpsandeepa@gmail.com, nadeeka-sumududissanayaka@gmail.com, tharindu@eie.ruh.ac.lk
Madhusanka Liyanage is with the School of Computer Science, University College Dublin, Ireland and the Center for Wireless Communications, University of Oulu, Finland. e-mail: madhusanka@ucd.ie, madhu-sanka.liyanage@oulu.fi

tracking, they can take measures to isolate this high-risk personnel. It ultimately slows down the disease progression, and reduces the total number of patients.

Conventionally, authorities tend to use manual contact tracking of positive COVID-19 cases. That method is not practical and inefficient in the long run since it needs the effort of many personnel, including health workers. It is also time-consuming and takes about three days per each new case [2]. Spending three days to identify a new case is useless when the disease is spreading rapidly. There are many existing approaches to automate contact tracing. However, there are some issues we observe in the existing solutions. As the first step, identification of all social interactions during the incubation period of up to 14 days is required. The identified close contacts should undergo social isolation. Verifying social isolation for these people with solely GPS-based solutions seems insufficient since GPS has less accuracy and does not work well indoors. Second, during our study, we noticed that there is a lack of related works regarding social interaction-based infection prediction methods due to the difficulty of tracking and accurately obtaining patients' past interactions. Third, healthcare workers pose a high risk of being infected with the COVID-19 virus since it is unlikely to identify the COVID-19 carrier without symptoms unless tested. Finally, there is a lack of proper mechanisms to identify correct groups of people who need testing for COVID-19. Using random methods or testing with less information will waste time, and resources. Besides, some of the patients may not be subjected to random tests and remain unidentified.

There are several contact tracing systems have been developed based on the Bluetooth Low Energy (BLE) [3]–[6]. However, none of these developments proposes mechanisms to provide social interaction-based infection prediction to compensate the above mentioned issues. Besides that, none of the current contact tracing systems have proposed an approach to optimize the mobile app's energy utilization even though the mobile applications usually run as background services. The advantage of our approach of having social interaction-based prediction is the increased accuracy in identifying the spread of the disease through social contacts. We show the accuracy of our predictions with the actual results under Section VI. Especially when considering the relaxation of travelling restrictions, there will be more interactions among people that could cause further spread of the disease.

In this work we propose a BLE-based social interaction

tracking system to address the aforementioned issues. For this, we use a mobile app to collect information about the nearby phones that use BLE. The app receives and records the proximity of other mobile phones and interaction duration based on the received signal strength. Then it uploads the collected information to a cloud storage with the timestamp and optional GPS location. We formulated a social interaction-based COVID-19 infection probability prediction algorithm on the server-side. Moreover, with the server and the developed mobile application, the system also facilitates real-time contact tracing if the user has enabled real-time data transmissions with the server. According to this option provided by design, a continuous contact tracing process will be carried out. In this case, both BLE and GPS data will provide real-time contact tracing for the system administrators. In addition to these implementations, we have also suggested an energy-saving approach associated with the mobile application by avoiding unnecessary power utilization associated with the mobile by sharing the total power requirement among various mobile users.

We designed an automatic alert generating mechanism to identify and notify the user about critical events. An example of such an event would be identifying high probabilities of being infected with COVID-19. Additionally a prototype implementation has been carried out of the solution and compared the validity of the proposed system with real world COVID-19 patient data sets.

The remainder of the paper is arranged as follows: Section II provides background information. The related work is presented in Section III. In Section IV, the system architecture and designed algorithms will be presented while in Section V, system implementation information are discussed. Section VI presents the experiment results. Finally, the paper will be concluded with conclusion and future works presented in VII.

II. BACKGROUND

This work's motivation is the COVID-19 outbreak in Wuhan city, China, and the epidemic situation occurred worldwide within a brief period. In this study, we paid attention to developing contact tracking systems to track the possible contacts exposed to COVID-19 patients. The system development implements the BLE as its primary wireless communication medium to perform the contact tracing functionality.

A. COVID-19

Emerging in late December 2019 from Wuhan, China, the Coronavirus disease 2019 (COVID-19) has caused a global pandemic within a few months. It is transmitted through droplets generated via coughing, sneezing, and speaking [7]. Therefore, it can spread during close contact between an infected and uninfected person by inhaling or entering the body via the eyes, mouth, or nose. The incubation period of COVID-19 is between 1-14 days. The median incubation period was estimated to be 5.1 days in [8] and 97.5% of those who develop symptoms will do so within 11.5 days of infection [8]. However, during this incubation period, one could spread the disease without showing any symptoms. Therefore,

fourteen days of active monitoring period is recommended to control the disease spread. It is a big challenge to prevent the disease while reporting new cases related to potential infectious sources, recognizing close contacts, and isolating them.

B. BLE

BLE is a low-energy version of Bluetooth specified in version 4.0. It recognizes as a modern Bluetooth technology developed by the Bluetooth Special Interest Group (SIG). Moreover, it is intended to support short-range communications [9]. BLE is a single-hop solution applicable to various use cases such as healthcare, consumer electronics, and smart energy and security. As mentioned before, several improvements of Bluetooth have been offered with BLE, and the raw data rate in BLE is 1Mb/s, and it had coverage up to a few tens of meters [10]. In BLE, the Received Signal Strength Indicator (RSSI) of the signal at the receiving end can approximate the distance between two devices. RSSI depends on distance and broadcasting power. To get an estimation of the distance, a simplified form of the relation between distance and RSSI is widely used [11].

III. RELATED WORK

Along With the emergence of SARS-COV-2(New Corona Virus / Covid-19) as a global pandemic, leading technology businesses and governments began developing contact tracing systems to discover suspected COVID-19 patient connections while protecting their privacy and data. In this scenario, they adhered to a number of principles in order to fulfill their objectives in relation to these advances.

Among them, Google and Apple have collaborated on a system with a contact tracing tool that works on both Google Android and Apple iOS operating systems in modern portable devices (smartphones). With this development, each mobile phone pings the other continuously via Bluetooth. In this situation, if two mobile phones are within range of each other and can receive Bluetooth transmissions, both will record the contacted mobile's ID [3]. Users must manually transfer the contact tracing records to the server, according to the developers' guidelines. This method will not work if COVID-19 patients refuse to report it to the appropriate authorities.

The Massachusetts Institute of Technology (MIT) has a similar development with many other researchers' collaboration. This system basically relies on Bluetooth signals which consist of random numbers (which sends as strings), likened to "chirps", and those other nearby smartphones can hear and remember these "chirps." In this case, the whole process is supported by an application installed within smartphones. This application can remember what was broadcast via the smartphone and what was received from the neighbouring handsets. Users will manually input the list of chirps into the server if they test positive for COVID-19, and alerts will be sent to the contacts who are at risk of infection [4]. This is similar to the technique described in [3].

Moreover, Singapore SGUnited, GovTech, Singapore, and the Ministry of Health implemented another system, namely
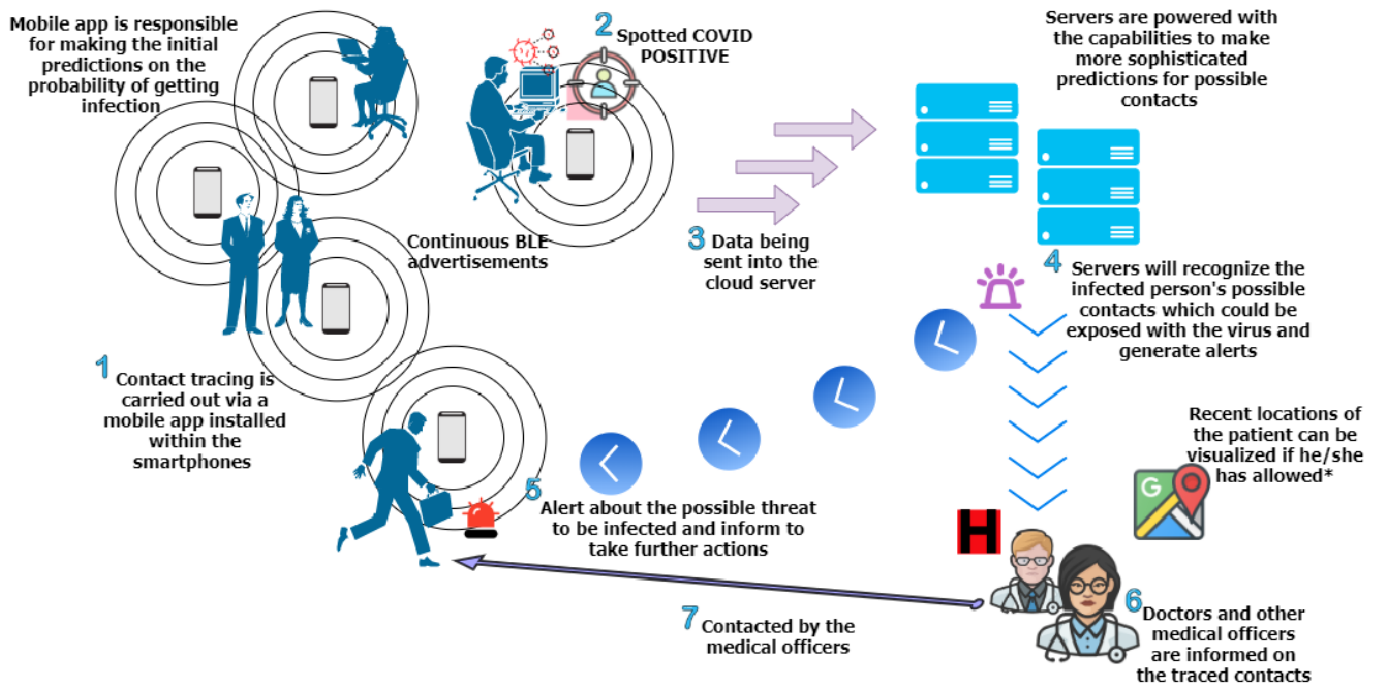
Fig. 1: Proposed System Architecture

"TraceTogether" for contact tracing of COVID-19 patients. Similar to the above applications, this also uses Bluetooth by following similar principles as mentioned in [3] and [4]. In this case, users of the system will also have to install a specific application on their smartphones. Once an individual has been tested positive for virus infection, certain users can choose to allow health authorities to access the data in the app to help identify possible contacts with higher risks [5]. Besides that, the government of Australia introduces a similar type of application, namely "COVIDSafe". In this case, the system is also based on Bluetooth. It provides the user with functionalities to perform manual uploads to the server containing possible high-risk contacts (if the user becomes positive for COVID-19). Even though this development doesn't discuss periodic changing of a user ID (to avoid unnecessary tracking of devices), it proposes a mechanism to store user data in an encrypted format to improve system security [6].

As stated in the preceding paragraphs, almost none of the developments described above have done real-time contact tracing and server-side algorithm implementation to improve accuracy of the whole system. Most of them support manual uploads of the traces into specific servers. Having real-time contact tracing is crucial in a highly contagious disease like COVID-19 since it has a higher infection and lower detection rates as shown in [12]. We include our methods of implementing server-side algorithms and make them scalable on-demand since an outbreak could appear randomly anytime. Then the servers should perform the required computations with increasing demand without an issue during such a critical time. Moreover, the servers' continuous participation/monitoring of the system and GPS tracking-related concepts are not discussed within the above implementations. Also, We noticed

that, even though the GPS-based tracking systems can lead to some privacy leakages of users. It is possible to avoid such situations by getting the users' consent and collaboration. In addition to that, data accumulated with GPS will be beneficial for conditions like high-sensitive contact tracing and efficient isolation. Not only that, the available systems do not indicate any probability measures to show the infection probability of exposed parties.

Furthermore, current developments generate some basic alerts based on general parameters like contact time and distance between users. But, according to our understanding, these alerts should be specifically tailored to be more helpful and gain maximum output. In this situation, the notifications should be customized for each user, taking into account their present medical issues as well as any chronic conditions they may be experiencing. None of the preceding systems, on the other hand, suggest tier-based detection. The tier-based approach's significance is that such a development can detect many direct and indirect contacts in multiple tiers of the person's connections. On the other hand, because the virus transmits in continuous chains from one patient to the next, this implementation is critical.

In contrast to the above implementations, the contact tracking system named "Aarogya Setu" developed in India [13]) has provided a competitive solution among the rest with hybrid BLE and GPS based contact tracing plus some unique characteristics like risk level predictions using the number of contacts and self-assessments based on symptoms and other relevant information like declared diseases, age, and gender of the user. Even though Arogya Setu seems to be a more sophisticated implementation, we found some issues associated with that implementation. This system lacks individual tracking protection as it has equipped with the randomly

generated static ID related to each user app. In addition to that, the system has not discussed continuous interaction with the server and real-time contact tracing and at the same time not providing any tier-based probability calculation method. Besides these, it has not addressed any power optimization methodologies. We designed a more sophisticated system based on these highlights and recognized flaws to minimize the challenges and weaknesses related with current COVID-19 contact tracing technologies.

With the work presented in [14], authors have discussed decentralized "Privacy-Preserving Proximity Tracing." They have carried out the developments to determine who has been in close physical proximity of an infected person while not revealing the contacted person's identity or contact location. Moreover, they have utilized many privacy-preserving measures by ensuring data minimization, preventing data abuse, and preventing tracking non-infected system users, and graceful dismantling. Our work has also implemented similar privacy protection mechanisms in many cases with some alternations and improvements.

Under the title of "Privacy-Sensitive Protocols And Mechanisms for Mobile Contact Tracing (PACT)", the authors have proposed a way to set transparent anonymity and privacy standards while maintaining and preserving civil liberties [15]. Moreover, this system will permit the adaptation of mobile contact tracing efforts. Furthermore, this work has introduced a third-party-free set of protocols and mechanisms to address its primary objectives. The researchers have also included the following capabilities: mobile-assisted contact tracing interviews, narrowcast messages, and privacy-sensitive mobile tracing via computing and communication technology.

It is possible to utilize BLE RSSI values/levels to estimate the distance between two mobile devices (i.e., users). The RSSI level is mostly determined by distance and broadcasting power. A simplified version of the relation between distance and RSSI is used [11] to get an estimation for distance. In [16], the authors provide detailed guidelines to estimate distances using RSSI value within a BLE system. They suggest a mechanism for calculating distances based on the RSSI level by altering the detection model. We also inspire this relation of RSSI and distance that was used in that work.

TABLE I shows a detailed comparison of existing contact tracing systems vs the proposed approach. According to that, it is clear that the proposing system provides a better level of uniqueness and a wider range of additional capabilities for contact tracing.

IV. Proposed Architecture and Algorithms

The Fig. 1 provides an overview of our proposed architecture. As shown in the diagram, there are three parties involved in the system, i.e., (i) smartphone users, (ii) cloud servers, and (iii) medical officers and health authorities. Each user installs a specific app on their smartphone to register with the system and use the service. During registration, the cloud server issues a random ID to the mobile phone, which updates over time to prevent traceability. After completing the registration process, each smartphone will automatically broadcast BLE

advertisements as a means of indicating its presence. Here, the BLE advertisement consists of only the random ID assigned to the mobile device. In addition, registered mobile phones continuously listen and record the advertisements from the nearby mobiles. In this case, the mobile app records two specific parameters: (i) the RSSI value of the received BLE advertisements and (ii) the duration of its contact with the mobile. Based on the user preferences and permissions, the mobile application securely records the mobile relay's GPS locations periodically. The app can also create initial risk level forecasts for each contact based on the above-mentioned personal data.

The data is then sent to the cloud server through a secure HTTPS connection, where it is used to do more refined second-stage predictions and contact tracking. The data transfer is done in real-time or on a periodic basis, depending on the user's preference and network availability. The health authorities can access server data and analyze it for detailed outputs on user movements and associated contacts if an registered user is infected with COVID-19. Under this scenario, we developed algorithms to help with the server function of extracting the links in a graph. Considering user privacy concerns, we set a threshold of 21 days period where the data is stored in the servers. As a result, medical officers can have a complete contact history dating back up to 21 days. We have created algorithms to calculate the risk levels linked with each of the patient's contacts and predict their infection risk. Finally, all close contacts receive messages based on their risk level, and the app instructs them on how to take additional steps to prevent disease spread, primarily by isolating oneself. Suppose the graph gives any nature of clustering in a particular region. In that case, notifications can be sent to the officers, such as public health inspectors working in regional areas, to prevent the disease from spreading beyond the cluster immediately. A map is also provided for the authorities to show recent locations of patients who have enabled their GPS tracking.

A. Calculation of contact distance using RSSI

The proximity between two mobile users is estimated via approximation because the distance between two users is a significant consideration. The RSSI of nearby mobile devices is the only measurement we have in this scenario. We use a simplified form of the relation between distance and received power of the signal [11], [37].

$$P_r(dBm) = P_{r1}(dBm) - K \cdot log_{10}(D(m)) \qquad (1)$$

where P_{r1} is the received power in dBm at 1 m, K is the loss parameter, and D is the distance between the receiver and the transmitter. Here, the value for K is obtained experimentally for an indoor environment. We take indoor environment by considering the worst-case scenario since the path loss for an indoor environment is higher than an outdoor environment [38]. In this regard, we calculate the maximum possible value range for K in an indoor environment. The value for K is, therefore, lesser in an outdoor environment according to the equation 1. For outdoor environments, if the user allows GPS in the app, it can be used as a measure in addition to the

TABLE I: COVID-19 Contact Tracing System Comparison

	1	2	3	4	5	6	7	8	9	10	11
Iceland: TRACKING C-19 [17]				✓							
Finland: Ketju [18], [19]			✓				✓				
India: Aarogya Setu [13]					✓		✓		✓		
France: StopCovid [20]			✓				✓				
Czech: eRouška [21]			✓				✓				
Germany: CoronaApp [22]			✓			✓					
Ireland: HSE COVID-19 App [23]			✓				✓				
Bahrain: BeAware Bahrain App	✓			✓							
Bulgaria: ViruSafe [24]				✓							
Israel: HaMagen [25], [26]				✓							
Malaysia: MyTrace [27]			✓								
North Macedonia: StopKorona [28]			✓				✓				
Maxico: CoviRadar [29]			✓				✓				
Norway: Smittestopp [30]	✓				✓		✓				
Italy: Immuni [31]			✓			✓					
Singapore: TraceTogether [32]			✓			✓					
UK: NHS COVID-19 App [33]		✓	✓				✓				
UAE: TraceCovid [34]			✓			✓					
Qatar: Ehteraz [35]					✓						
Poland: ProteGO [36]			✓			✓					
Our Development	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

1. Real time data transmission to the server and continuous contact tracing.
2. Switchablity between manual and automatic data transmission modes between smartphone and cloud.
3. Availability of Bluetooth based tracing.
4. Availability of GPS based tracing.
5. Bluetooth and GPS based hybrid contact tracing.
6. Tracking protection with periodically changing random ID.
7. Anonymous ID assignment for user.
8. Algorithm-based infection probability prediction.
9. Specifically tailored risk level prediction and alert generation for users (i.e considers age, other illnesses, etc.)
10. Tier based expansion for contact tracing.
11. Power optimization algorithm.

BLE based distance approximation. Here, the RSSI values and distance recordings were taken from 1 meter to 15 meters of distances in the intervals of 1 meter. We obtained RSSI readings 30 times in each position within that range. The values of K lies between the range 18 and 22 when applying the experimental average values to the above equation 1.

B. Energy Saving Algorithm

Even though BLE-based systems provide higher-level energy saving than the legacy Bluetooth-based systems, this work proposes a further power-saving algorithm to cut unnecessary power consumption within the system operation. We recognized that the measures taken via this power-saving algorithm would be beneficial under the mobile application's long-duration usage. Furthermore, with the suggesting algorithm, the primary intention is to limit the users' power utilization by sharing the total power requirement among different users evenly for a certain period. However, there is a trade-off between the system's accuracy and energy-saving when implementing this kind of algorithm. In addition to the proposing facts included in the following section, several theoretical derivations and simulations are contained within Section VI-C

to derive some parameters associated with the algorithm. The proposed algorithm can be recognized as a protocol based on two major concepts,

- Cluster-based contact tracing for idle (or restricted movement) users.
- Short duration communication with servers.

As indicated from the 1st point in the list, this algorithm is primarily developed for idle users (e.g., users who are working within an office environment, sitting in a bus stop, etc.) and users who perform a limited motion within a certain location (e.g., payment queue in a bank, etc.) for a certain period.

With this algorithm, users are divided into clusters. Each of these clusters consists of a maximum X number of users (X is a variable and refer to Section VI-C for more information). All of these mobiles are under the 'idle' or 'restricted movement' condition. Mobiles can automatically detect this via their built-in sensors. However, optionally users are provided with the capability to select the idleness using the smartphone application to indicate their idle state manually or confirming the notification that may be pushed after detecting the idle state via the smartphone sensors. Even though the idleness could be detected via the sensors, the constrained movement may

be manually confirmed by users to enable the energy-saving approach. After confirmation of the inactivity or constrained movements, the mobile application can automatically enter the energy-saving mode by following the special energy-saving algorithm accordingly.

1) Cluster Operation: Under the clustering process, the distance between each node within the cluster is limited to a maximum of 8m since it was recorded as the maximum distance in which the droplets can travel over the air from one infected person to another [39]. Moreover, one mobile can only be a part of a single cluster in this process, and each cluster is isolated from the others. With this clustering process, a single idle mobile (e.g., A) can be taken into a clustering agreement with $X - 1$ other nearby idle mobiles and form a cluster. The overall clustering process can be summarized as follows.

- First, as described above, a mobile detects its idle condition via the sensors. It pushes a notification to the user to confirm the idle state (This can also be performed automatically according to the user preferences). After the confirmation, it will advertise its idleness along with its random ID. (Note: The random ID is the periodically changing anonymous ID assigned to a particular user).
- If the idle indication is received by another idle device nearby, these two can get into a cluster agreement. In this instance, the distance between the two mobile is calculated using RSSI to ensure the 8m criteria has been satisfied.
- One mobile will contact the server during the cluster agreement and download a cluster-ID (valid only for the session). Up to this point, there are only two participants in the cluster, and the cluster-ID will be broadcasted as (cluster_ID, user_no, $X - 2$) - Here, the user_no is a random number assigned for each user in the cluster to identify each mobile within the cluster. $X - 2$ is the available number of vacancies within the cluster.
- The notable fact associated with the cluster operation is only one mobile performs the advertising and scanning work on behalf of the cluster while other mobiles remain under 'freeze' condition. However, after a certain period, this job will be transferred to another mobile within the cluster, and the previously functioned mobile enters the 'freeze' condition accordingly (Fig. 2). In this case, the only message transmitted by the delegated mobile is the advertisement (cluster_ID, user_no, $X - 2$) while scanning for the broadcasts gets from the mobiles outside the cluster.

At this point, three keywords associated with the cluster operation has to be introduced as follows,

- Phase: The time duration where a single device performs its advertising plus scanning operation (t minutes; where t is a variable).
- Session: Duration, where all the mobiles within the cluster go through a single-phase ($t * X$ minutes - This can be varied according to the number of participants within the cluster).
- Delegated Mobile: A mobile currently functioning in both advertising and scanning modes (i.e., Mobile, which is currently going through its phase).

Especially at the end of each phase, each mobile within the cluster performs a broadcast to notify others about his/her presence within the group.

When 3rd device comes into the proximity of the previously formed cluster (the new device should be in the idle mode as well),

- The new mobile will identify there is a vacancy for the existing cluster by reading its broadcast message. However, it has to wait until the moment where the next phase of the cluster initiates.
- At that point, the new device will take the chance to connect with the cluster. In this case, the distance between each of the clustered devices and new devices will be calculated. Afterward, the new device's joining request will only be processed if all the calculated distances do not exceed 8m (can be detected via RSSI). Or else, the new device has to be connected with another cluster.
- This process will continue until all the vacancies are filled within the cluster.

Furthermore, at the end of each session, all the mobiles will send a broadcast message to synchronize the collected data during the cluster operations (e.g., advertisements received from the out-of-cluster mobiles or external clusters). As a result of these broadcast messages, all the mobiles will synchronize their databases accordingly. Any missing data of an outside user or the cluster will get updated in this synchronization phase. Moreover, the distance between each node of the cluster must calculate at the beginning (the moment where the mobile device joined with the cluster) and the end of the session. Then, two values will be averaged to take the average distance between each node. If someone escapes from the cluster early, that person's data will temporarily be stored in a cache. Then, the data can be synchronized via the server at a later moment using the cluster-ID. On the other hand, if device sensors record a significant mobile movement within the cluster, it will automatically exit from the cluster operation.

2) Case With Multiple Nearby Clusters: As mentioned in the above section, any number of clusters can operate as isolated entities in the same environment. These clusters can be geographically mixed, as indicated in Fig. 3, but they should logically be separated into distinct clusters accordingly. Each cluster broadcasts the user's cluster-ID, the user_no, and the available number of vacancies via the delegated mobile under the cluster operation. As we consider nearby clusters, all of them may receive advertisements from other nearby clusters. These advertisements may not be relevant as the distance would be too large to cause infection. As described in Section VI-C, we prefer short-duration phases under the cluster operation. There is a higher chance of having every mobile in a cluster perform broadcasting during a short period.

When new mobile comes into a multi-cluster environment and requests to connect, two conditions have to be considered. First of all, there should be vacancies in the cluster to be joined. Secondly, if the new mobile device receives advertisements from multiple delegated mobiles from different clusters. It should connect with the cluster which sends the message
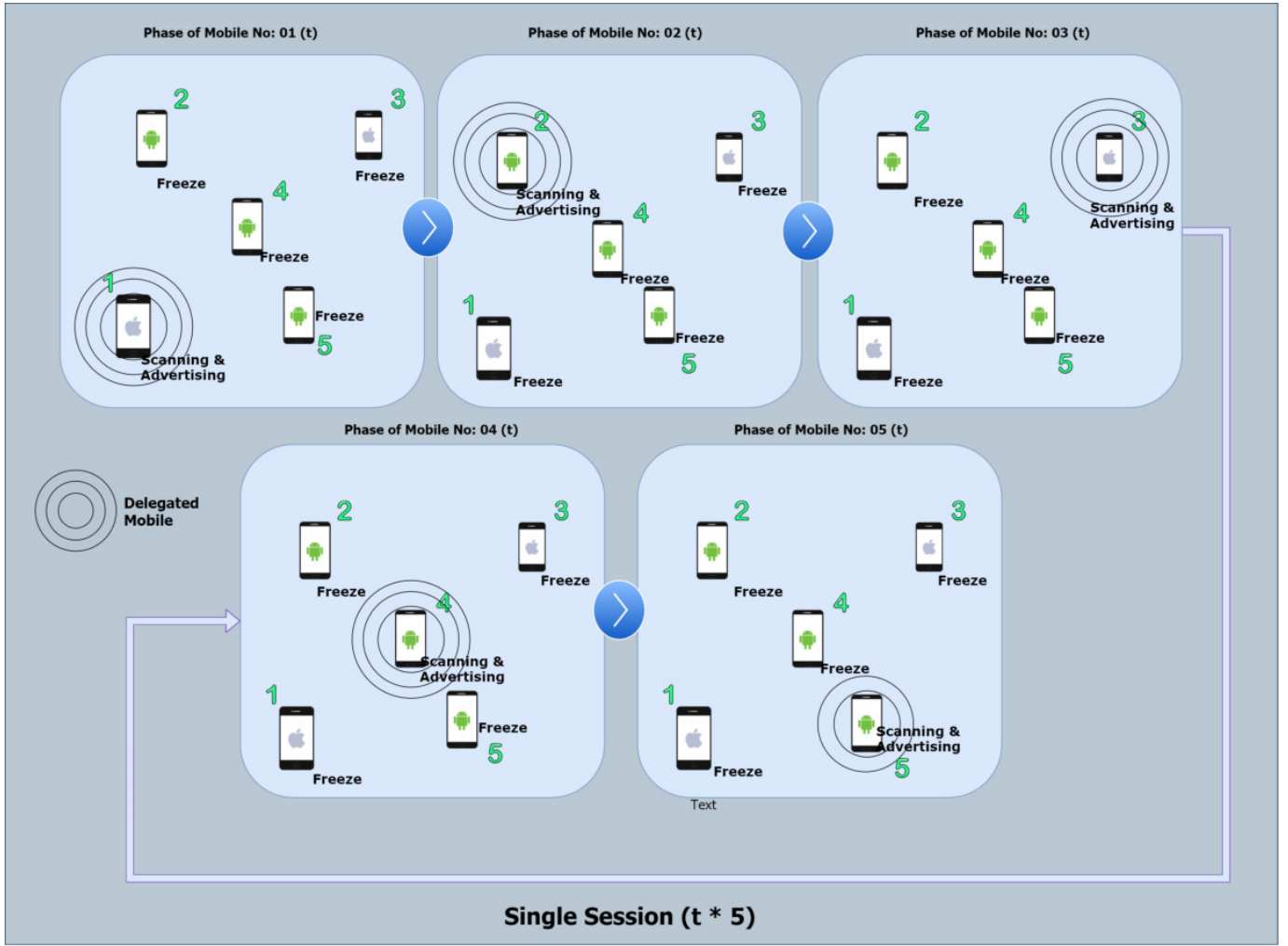
Fig. 2: Cluster Operation - Overview

with the highest RSSI value. The first condition should have to be satisfied in order to go with the second condition completely. On the other hand, if more than X (maximum number of mobiles devices which can be clustered in) have received advertisements from a single delegated mobile, we come to a situation where one of the new mobile device (which want to be clustered) initiate to form a new cluster with the same proximity. As mentioned before, these clusters can be geographically mixed but logically isolated.

At the end of each phase, both advertising and scanning modes will enable in every mobile. Therefore, all of them receive advertisements from nearby clusters. Simultaneously, advertisements sent from these mobiles will receive by delegated mobiles in the nearby clusters. If any data get lost during this process, it will recover during the database synchronization phase at the end of the session. Furthermore, a mobile in a particular cluster may receive multiple advertisements from the mobiles in a nearby cluster. Therefore, TABLE II's approach can calculate the distance between each mobile in separate clusters with the server's aid (Note: The user numbers are temporary IDs that will only be valid for a particular session).

As described above, we can perform multi-cluster commu-

TABLE II: Example Distance Calculation Among Mobiles Under Multiple Cluster Operation

Cluster-ID	User_no_1	Average RSSI from user 1	Average distance to user 1
	User_no_2	Average RSSI from user 2	Average distance to user 2
	User_no_3	Average RSSI from user 3	Average distance to user 3
	User_no_4	Average RSSI from user 4	Average distance to user 4
	User_no_5	Average RSSI from user 5	Average distance to user 5

nication via the cloud database, which helps to synchronize the data between the clusters. With this approach, even if there is a COVID-19 infected person in a nearby cluster (but not within a certain cluster), the other nearby cluster is informed, and their databases being updated accordingly. This will support the tier-based detection scenarios as described in the following sections.

Note: If the above approach is unsuccessful, the approximation approach presented in the following section (considering the worst-case scenario) can be utilized accordingly.
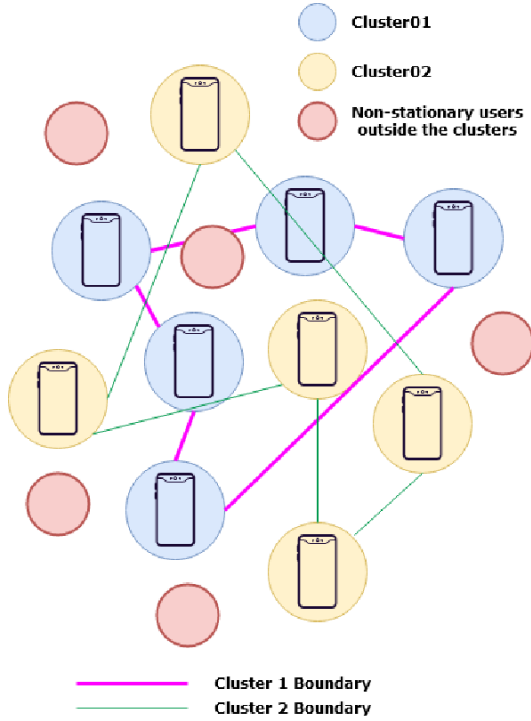
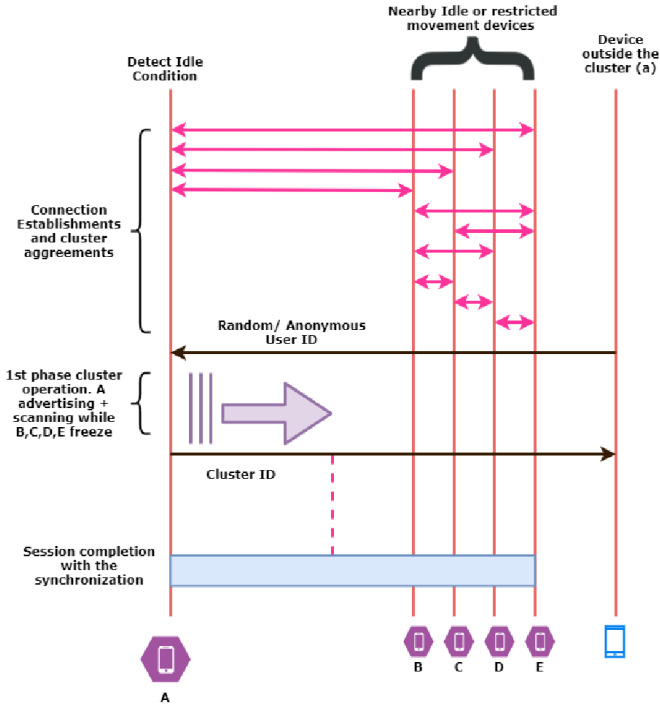Fig. 3: Operation With Multiple Clusters

Fig. 4: Message Flow Diagram

3) Distance Calculations from Out-of-Cluster (Dynamic or non-dynamic) Mobile: As mentioned above, all the clustered devices are assumed to be nearby devices. Therefore, all of them may be influenced by a nearby out-of-cluster device, and their impact can also be valid for other mobile users. Here, out-of-cluster users can be divided into two major parts such

as,

- Out-of-cluster users who are the members of another cluster (can expect lesser movement, non-dynamic users).
- Dynamic users who are not a member of any cluster (can expect dynamic movements for a short period).

However, under the cluster operation, instead of sending all the IDs individually, one mobile (A in the message flow diagram, Fig. 4; delegated mobile) will send the session ID (cluster-ID) to the nearby out-of-cluster devices to save more energy and processing power. Therefore, when considering the distance calculation between the clustered mobile (e.g. C, Fig. 4) and an out-of-cluster mobile (e.g. a, Fig.4), a is receiving only the cluster-ID and the user_no. However, this external mobile can index the server database with this cluster-ID regarding the information about that particular cluster (requires an internet connection). With the data retrieved from the server, the distance between each user will be calculated according to the worst-case scenario. In this case, the minimum distance between the external user and the user within the cluster will be calculated accordingly.

E.g.
External User - a
Broadcasting User - A
Inactive mobile in the cluster - C
Calculated Distance from a to A - 5m
Distance form A to C = 2m
Then the distance from a to C under worst-case = 5-2 = 3m
If the worst-case distance recorded something less than 0m, the absolute value is taken (e.g. 3-5 = -2m $\Rightarrow$ 2m)

C. Infection Probability Prediction Algorithm

We offer a method for calculating the likelihood of infection for a given user. When considering the chance of spreading an infection from one person to another, we assume that two key criteria would have the most impact on the algorithm: the distance between them and the time they were in contact.

We suppose that the likelihood of contracting the virus varies with distance in an exponential manner by considering the rapid increase of getting the infection when two people are getting closer physically. This is shown by the equation 2 as follows:

$$P_d(x) = e^{-nx} \qquad (2)$$

where n is a positive constant, and x is the distance. For example, in [39], the probability of getting the infection at a 4 meters distance is assumed to be 5%, and the value of n is calculated accordingly.

The probability variation with the contact period is taken as,

$$P_t(t) = 1 - e^{-mt} \qquad (3)$$

where m is a positive constant, and t is the period of contact. According to this equation, the chances of getting the infection increases with more time in interaction. In our simulations, based on the [40], 95% confidence of getting the infection is taken as 10800 seconds (3 hours) of the contact period.
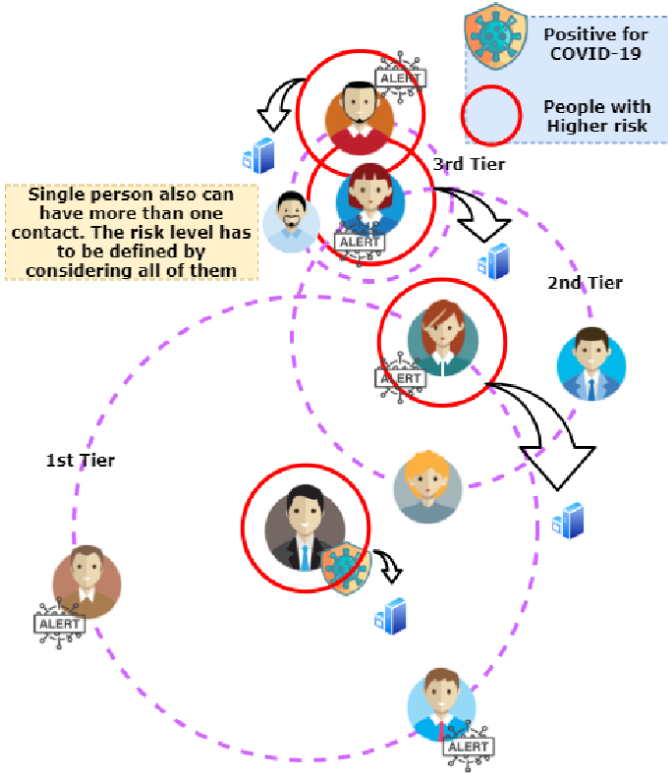
Fig. 5: Example Contact Tracing Process for a Single Sequence up to 3 Tiers

The simple scenario for contact tracing in a single sequence is described in Fig. 5. As illustrated, for two contacted people a and b where a is in tier r and b, is in the tier $r+1$, (where b is an indirect associate of a), their contact distance d_{ab} and time t_{ab} are considered as independent from each other. Thus the two probabilities are multiplied to get the probability of having b being infected by a. We denote this probability as $P_i(b,a)$. Also, the total probability of getting the infection P_i to the person a affects the probability of getting the infection to b. They can be represented as follows:

$$P_i(b,a) = P_i(a) \cdot P_d(d_{ab}) \cdot P_t(t_{ab}) \tag{4}$$

Apart from distance and time, we took into account a variety of other aspects, such as the user's medical issues, to determine the score, provided the user is ready to disclose these personal facts. If the contribution of a factor h to the probability is denoted as P_{c_h}, by considering k number of such factors, the probability $P_i(b,a)$ of person b getting infected by the person a, can be denoted by the following equation:

$$P_i(b,a) = min(P_i(a) \cdot P_d(d_{ab}) \cdot P_t(t_{ab}) + \sum_{h=1}^{k} P_{c_h}, 1) \tag{5}$$

The value of k is dependant on the level of refinements that one would want to achieve. The primary two considerations in our algorithm are, as mentioned, the distance and the time measures, which have the highest impact on the overall probability. When using the algorithm, it is up to the users, such

as governments and organizations, to consider other factors and perform necessary experiments to identify their impact on the overall probability. With more parameters considered, the probability score would increase according to the significance of the factor. For a use-case, we considered some Risk Factors (RF) based on several considerations, where they contribute to the risk of getting the infection. Here, we use the case mortality rates presented in the [41] to derive the values for the P_{c_h}. Age, comorbidity, and gender are the three most important factors to consider. People with chronic conditions and those over the age of 60 are more susceptible to the disease due to their weakened immune. The higher COVID-19 death rates in the elderly population support this fact. As a result, we add this extra weight to the previously computed probabilities based on distance and contacted duration, in order to create more tailored results for each user. Moreover, the risk factor calculation for base case scenarios is available in TABLE III with an assignment of 0.01. In addition, TABLE III shows the RFs related to different situations, which are calculated as relative variations of the base case scenario.

TABLE III: Base-Case Scenarios and Relative Risk Factors (RF) [41]

Age Group	RF	Comorbidity Condition	RF
0-9	0.010	Healthy (No-Comorbidity)	0.0100
10-19	0.010	Cancer (Any)	0.0622
20-29	0.010	Hypertension	0.0667
30-39	0.010	Chronic Respiratory Disease	0.0700
40-49	0.020	Diabetes	0.0811
50-59	0.065	Cardiovascular Disease	0.1167
60-69	0.180	**Gender Consideration**	**RF**
70-79	0.400	Female	0.0100
Age =>80	0.740	Male	0.0165

As the person b can have q number of multiple connections from the previous tier, we get the total probability of being infected as b, $P_i(b)$, which is the minimum of either the sum of all probabilities from each connection or 1. This makes the maximum possible probability equal to 1. If the probability reaches 1, it implies that the person is infected.

$$P_i(b) = min(\sum_{j=1}^{q} P_i(b,j), 1) \tag{6}$$

V. IMPLEMENTATION

We develop BLE and location-based prototype mobile application to show the practicality of our proposal.

A. Mobile App

We develop a mobile app for Android mobiles using Android Studio 3.4. The mobile app interface is shown in Fig. 6, it is the interface through which users interact with the system, and it can also be used to track user interactions. It performs many tasks, including user registrations, broadcasting BLE advertisements, and sending user data to the server. The app records the user's contact information during the registration phase, such as mobile number, address, and email details. At the same time, users can supply some health-related data (e.g., chronic conditions) to receive more accurate and
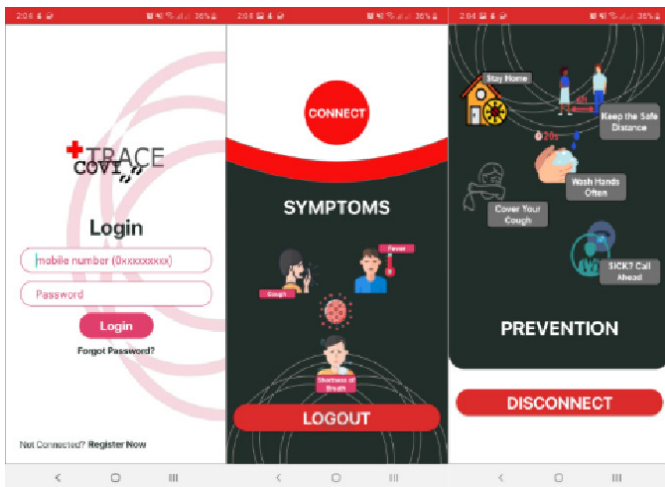
Fig. 6: Developed Mobile Application

especially designed notifications, albeit it is not mandatory. After completing the registration process, the user can log in to the app and enable the automatic functioning mode. The app will then operate without any user interaction.

Additionally, users can enable or disable privacy features like GPS tracking, personal information, health data, etc. Here, the GPS data gets only recorded when the user allows for this feature, and even if the user enables permission, it is only stored locally in the user's mobile. Then, the infected person can anonymously contribute to sending the saved locations to help authorities warn the public about the patient's recent locations without revealing the patient's identity. There are two modes for BLE in the app: advertising mode and BLE scanning mode operations. After each scan, the RSSI values of surrounding mobiles (which convert to distance values), the user location, and associated timestamps are captured and saved in the app memory.

Additionally, the app records the overall contact period with each of the nearby mobiles. Once the internet is available, or during the user's allocations, the app transmits data via a secured web socket to the cloud server. Here, the user can select one of the three options. The first option is real-time (users have to enable this to perform real-time (or continuous) contact tracing mechanisms). The second is to send them periodically, and the third is to perform manual transmission according to user preferences.

B. Web Application

The "Java Spring Boot framework" is used to implement the server side. The back-end is made up of microservices using Application Programming Interfaces (APIs). When more user traffic is available, this model helps to scale up the services. For the front-end, we use the Angular framework to view mobile users' data by authorized administrators. This application consists of a login portal where only authorized users get access to it. After successful login, the administrator can input any user's phone number. They can then check the position and details of nearby persons for a specific time period

as shown in Fig. 7. They may view the locations of nearby users' details for each pin on the map using Google Maps (Fig. 7c). The administrator can also use a network graph to visualize these connections over time to gain more insight into the interactions (Fig. 7a, Fig. 7b). We allow the graph to view up to 3 tiers of information about direct and indirect contacts for demonstration.

Authorities can generate alerts via email and SMS message services after locating the contacts. People at higher risk can be automatically notified. They can then be self-quarantined, and get them self tested using a PCR performed in order of decreasing risk.

C. Security and Privacy

In our work, we considered the security and privacy of the users since it would be a critical aspect regarding user data. For Location privacy preservation, we provided the option for the users to choose if they want to provide their GPS location. If not, the BLE-enabled location will be utilized. We used a lightweight yet robust Advanced Encryption Standard (AES) encryption technique to encrypt data from the user's mobile application to establish End-to-End (E2E) confidentiality between the mobile device and the cloud server. A secured HTTP connection is made with the users on the server-side to connect with the system, and all the communicating data is encrypted.

VI. EXPERIMENTS AND RESULTS

A. Analyze the Performance of Infection Prediction

To identify the performance and verify the usability/validity of the algorithm, some system simulations have been performed accordingly. For comparing the simulations with a real data set, a COVID-19 patient infection dataset [42] was selected, and simulations were performed. The starting date of the selected data set, indicated as 22nd January 2020. The simulation results are shown in Fig. 9. We selected several provinces in China, Vietnam, Taiwan, two states of Australia, and two islands of the United Kingdom, where the disease has spread. These locations had steady control in relatively smaller populations, such that disease is assumed to have been controlled, and all the cases were detected within the selected period. We selected many regions in different parts of the world to verify if our model fits the locations' real data disregard. The simulations of the proposed model, both with and without considering personal details, show that our approach can be utilized to obtain similar results to the real statistics. Therefore, the selected locations resemble our graph-based approach with a fixed number of nodes. A graphical representation of such a graph with 100 nodes is shown in Fig. 8.

The users are represented with nodes, and the connections among the users mean with edges in the graph. Each node within the graph contains the infection probability of a particular user. We developed two models for each selected region or country with the exclusion and inclusion of a user's personal details. Furthermore, we took pieces including chronic medical conditions, gender, and age into account. In this scenario, the gender is chosen at random and distributed
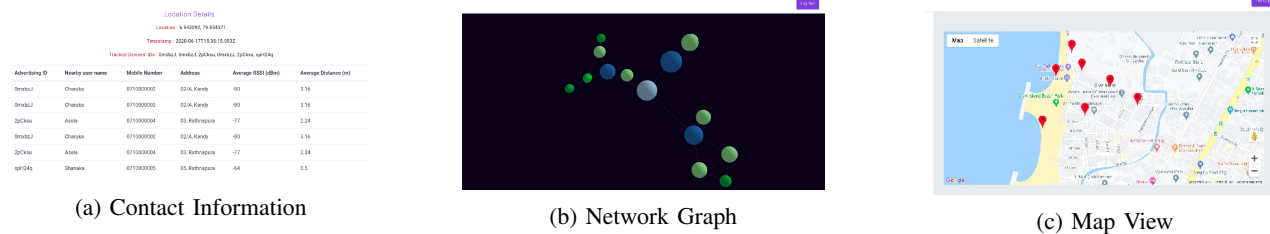
(a) Contact Information

(b) Network Graph

(c) Map View

Fig. 7: Web Application
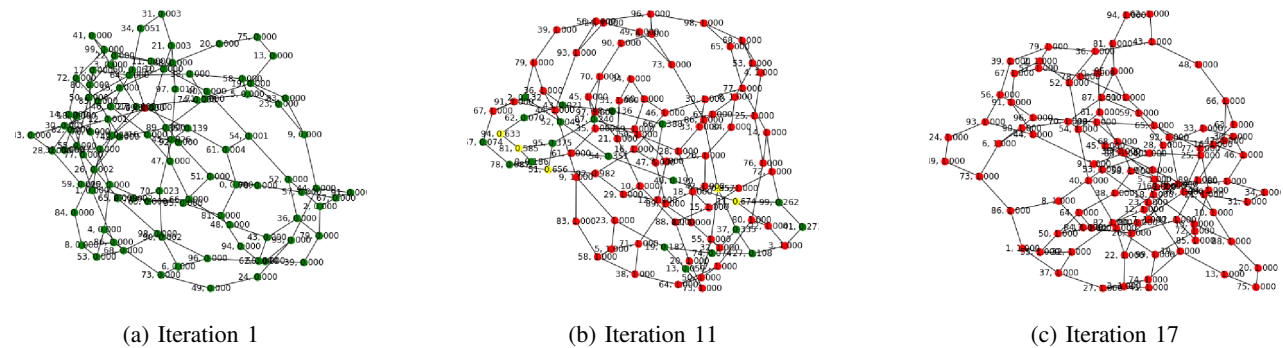

(a) Iteration 1

(b) Iteration 11

(c) Iteration 17

Fig. 8: Graph Model for 100 Nodes at Different Iterations of Infecting Nodes

evenly over the graph. When it comes to age, a triangular distribution is approximated so that the population's average age is considered the maximum in the probability distribution. Chronic disease conditions are deemed to rise with age, and a maximum likelihood of two diseases per individual with twice or more than the average age is specified.

An edge in the graph model contains the average distance and duration of contact between two nodes over a while. As a result, the created graph reflects a specific time, along with the connections.

The graphs are formed with a set number of nodes, and edges are generated at random. This randomly generated number of edges is set to have a minimum of one edge and up to a maximum of $n - 1$. In this case, n is the total number of nodes initialized in the graph. In addition, each edge has values for distance and contact period that are randomly initialized. The likelihood of a node becoming infected is initially set to zero. Then, by selecting a node at random, a new patient is introduced into the produced graph. All other nodes in the graph update their infection probability using our proposed algorithm for infection probability. The technique is then repeated for multiple iterations until all nodes have been infected.

We simulated the insertion of one new patient randomly in the graphs for each iteration, as suggested by the method. One iteration is deemed to be equivalent to one day of real-world data in this example. The total patient count obtained from both simulated and real data was then plotted. The number of iterations in all simulations is smaller than the total number of nodes in the graph, as seen in Fig. 9. The reason for this observation is that the probability of infecting healthy nodes increases with each iteration, and some may become

infected automatically. This observation resembles how the virus spreads from one patient to a healthy person.

In each of these simulations, the maximum number of node connections shows the maximum possible edges that a node can have. We simulated for maximum node connections ranging from 2 to 5 in each graph in Fig. 9. These graphs show that when the maximum possible edges are reduced, all the nodes in the graph are infected within fewer iterations. With this result, we can conclude that self-isolation and social distancing-related measures are important to reduce the speed of disease spread.

For each graph that has considered the personal details, we selected the best number of maximum node connections based on the curve having the highest Pearson correlation coefficient with the real data curve. Because the edges are produced at random, we performed simulations for each of these curves, infecting all of the nodes for 50 iterations and calculating the mean patient counts. The Pearson correlation is measured for the mean simulated data. The best correlations and their respective maximum node connection counts are shown in TABLE IV below.

The following sections provide further observations that we made from simulations in each country.

1) China: In China, we selected four regions, Hainan, Shaanxi, Shanghai, and Sichuan provinces, to simulate total patients count with actual data. Then, we modeled graphs for each area with the same number of nodes as the total number of patients detected within a period. This period may differ as the time taken to control the disease may vary from one region to another.

2) New Zealand: The data set of New Zealand shows a rapid increase in the patient count after about five patients
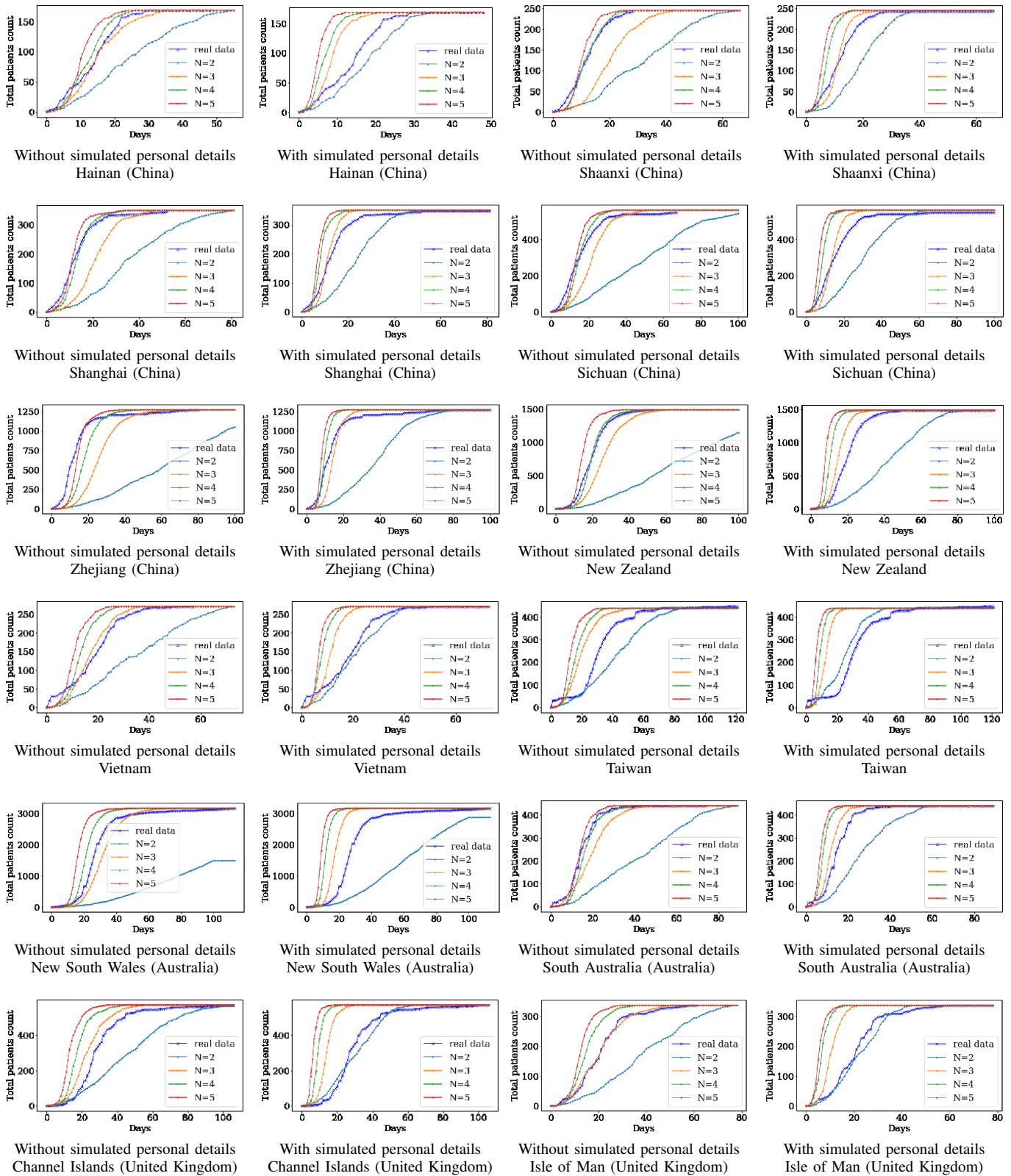
Fig. 9: Total Patients Count with Changing Maximum Graph Node Connections for Different Regions and Countries

were discovered. The reason for this is that a single case introduced to a new place does not always result in an epidemic. Still, multiple introductions can result in an epidemic [43]. Therefore, we started the patient count of real data from 5 patients. The comparison without simulated personal details has provided closer results to the actual data at the maximum connections of 4 for a node.

3) Vietnam: Fig. 9 also shows the comparison of simulated data with the recorded patient count in Vietnam. Here, the patient count was taken, starting from 16 patients, as the time it took to reach 16 patients is high compared to the rapid increase in the patient count afterward.

4) Taiwan: The comparison performed in the Taiwan dataset was initiated at a patient count of 30 as in the initial stages, the disease's progression was slower.

5) Australia: Two states in Australia, New South Wales, and South Australia, are selected to compare the disease spread. In New South Wales, the maximum node connections are between 3 and 4 without personal details. When added personal data, the simulations shift, such that the actual data is in between 2 and 3 connections. South Australia has about four node links without considering the personal details, and with the details, the node connections get closer to 3 connections. This change may occur due to increased overall probability with personal data and approaching the infection state faster. It would cause a shift in the curves to reach the whole infected state with lesser iterations.

6) United Kingdom: Data from two islands of the United Kingdom, the Channel Islands and the Isle of Man, were selected to compare with the simulations. In both cases, we can observe a relatively high shift in the curves with personal details.

TABLE IV: Results Summary for Graph Simulations for Different Countries

Region/Country	Selected Maximum Node Connections	Pearson Correlation Coefficient	Selected Maximum Node Connections with Personal Details	Pearson Correlation Coefficient with Personal Details
Hainan (China)	3	0.991	2	0.983
Shaanxi (China)	4	0.994	3	0.987
Shanghai (China)	4	0.998	3	0.993
Sichuan (China)	4	0.991	3	0.978
Zhejiang (China)	5	0.982	4	0.983
New Zealand	4	0.997	3	0.963
Vietnam	3	0.997	2	0.993
Taiwan	3	0.942	2	0.987
New South Wales (Australia)	3	0.984	3	0.900
South Australia (Australia)	4	0.998	3	0.983
Channel Islands (United Kingdom)	3	0.986	2	0.992
Isle of Man (United Kingdom)	3	0.998	2	0.987

From the TABLE IV, it can be observed that all graphs have a high correlation value of more than 0.9 for the best-selected maximum node connections. Out of them, 4 locations have a higher correlation with the actual data when considering personal details than those that don't consider the personal details. But in other cases, higher correlations can be observed in the graphs that did not consider the personal information. As mentioned, this may occur because the nodes increase probability due to adding personal details to the overall probability. It would then result in infecting the nodes in the graph at a faster rate. The application of personal information may depend on the region's spread, where the infection is spreading

faster. If the infection spread is slower, confidential personal details may not be required from the public. Social distancing also needs to be considered as the maximum connections that a node can have may change when switching between them. Thus, it would be essential to consider both cases and select the more accurate outcome based on the region, social distancing, and the spread.

B. Threshold Probabilities for PCR Testing and Self-Isolation

We simulated graphs with 250, 500, 750, and 1000 nodes to find a threshold probability for PCR testing and self-isolation with this model. The number of connections among the two nodes was varied from 2 to 10 in each of these graphs. The nodes were made infected, selecting one random node at each iteration until all nodes get infected. Mostly, the nodes get infected due to nearby ones and increasing their probability, as suggested in our algorithm. If a node receives the infection at n^{th} iteration, we recorded the probability of the node at $(n-1)^{th}$ iteration. The mean probability and one standard deviation were plotted for each of those graphs, as shown in Fig. 10.

The graphs show that the lower the number of connections, the lower the probability of being infected. This probability increases with the maximum number of links that a node is allowed. When considering lesser mutual connections, the nodes often get the infection randomly, and thus their probability before contracting the disease is lower. Also, the impact of getting infection to a nearby node is high and more down in distant nodes; this resembles a real-world situation. Therefore, when social distancing is imposed, the threshold probability of issuing a warning could suggest as 0.3 - 0.4, as indicated by the mean of the graphs. Afterward, PCR testing and self-isolation measures can be done with the upper bound of standard deviation at a probability of 0.6 - 0.7.

In the case of lesser social distancing restrictions, the chances are to have more interactions among people. Thus, there would be more additions to the probability of getting infections with indirect contacts. Therefore, as suggested by the simulations, the threshold probability is higher. This would be important as generating false alarms at lower thresholds can reduce the system's effectiveness, and users may lose trust. From the simulations, it is suggested to have a threshold warning probability of 0.4 - 0.5. The self-isolation and PCR testing can be done when the possibility reaches the upper confidence level of standard deviation, which can be in the range of 0.7 - 0.8.

C. Performance of the Mobile app

In addition to the previously mentioned developments, power utilization of the mobile application has to be taken into account. Since the developed mobile application is running as a background service in the smartphone, high power efficiency is essential. However, most of the similar research done under the title of "Contact Tracing" associated with the COVID-19 pandemic has not addressed the power optimization measures related to the system. They have solely relied on the inherent

(a) For a Graph of 250 Nodes

(b) For a Graph of 500 Nodes

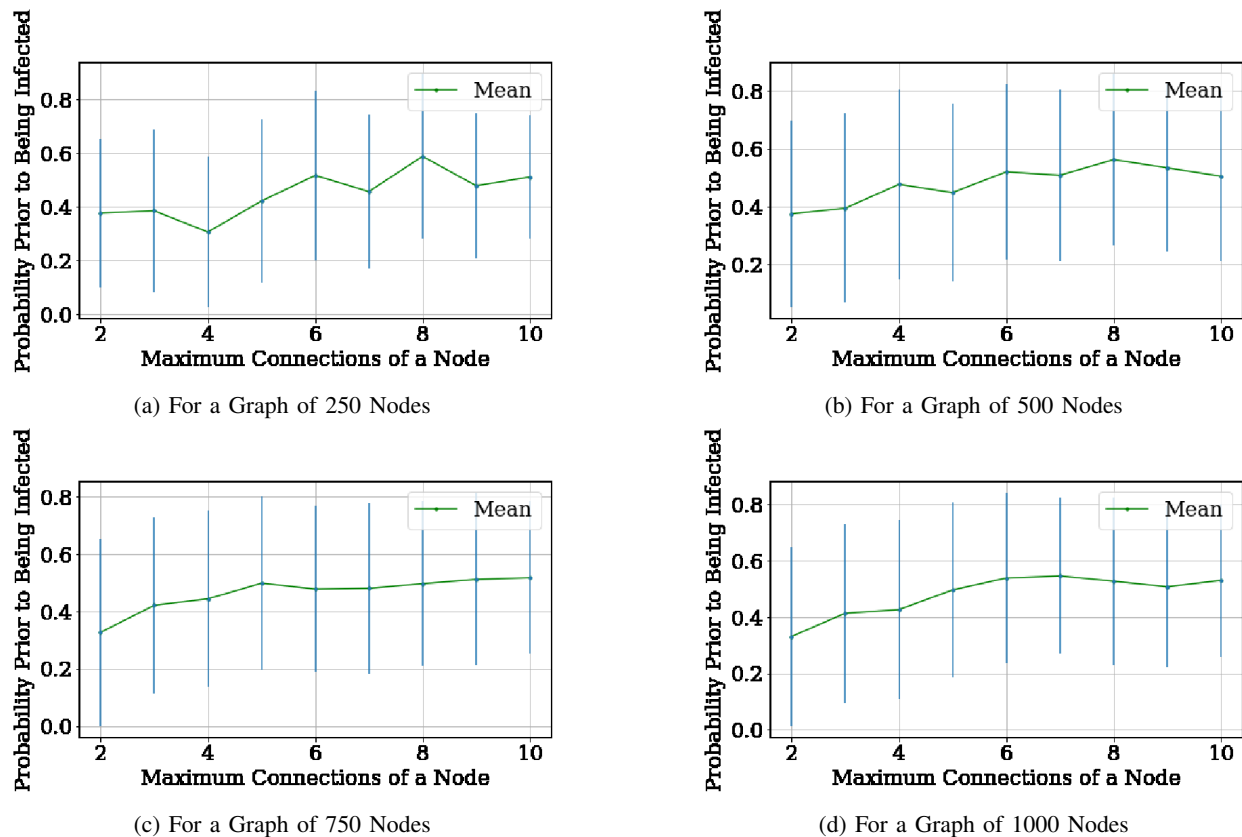(c) For a Graph of 750 Nodes

(d) For a Graph of 1000 Nodes

Fig. 10: Probability Prior to Getting the Infection to a Node in Graphs of 250, 500, 750 and 1000 Nodes

power-saving characteristics of the BLE. However, our energy-saving algorithm is expected to achieve an energy-saving up to 90% theoretically with the proposing total energy requirement sharing strategy (Under idle or restricted movement of the users).

For this experiment, we make the use of two android smartphones, and the obtained results are indicated in TABLE V.

TABLE V: Mobile App Power Utilization

Mobile	Battery Capacity (mAh)	Total Memory (RAM)	App Run Time	Energy Consumption (mAh)	Energy Consumption as Battery Percentage(%)	Memory (RAM) Usage (MB)
Samsung A20	4000	3GB	1 hour	8	0.2	33
Samsung M20	5000	4GB	1 hour	10	0.2	36

According to the observations indicated above, it is obvious that the mobile application would run with considerably lesser power consumption even though it runs as a background service all day. In this case, the estimated power consumption for a 24 hour period is 192 - 240 mAh. It is only about 4.8% power consumption compared with the total battery capacity of modern-day smartphones (4000mAh/5000 mAh of battery).

In addition to the above observations, with the use of the data present in TABLE V, we have derived the following results that clarify the features indicated in the proposed protocol in the section above. Fig. 11 indicates the impact of cluster size on the app's overall power consumption. In

this case, the phase duration is kept for a 1-minute duration while changing the cluster size from 2 to 20. Furthermore, the simulation has extended to different session duration as 1 hour, 2 hours and 3 hours as indicated on Fig. 11a, 11b and 11c respectively.

According to the results presented in Fig. 11 it is evident that with the increase of cluster size, the overall power consumption also becomes lower. However, the abrupt decrease of power utilization neutralizes in each case approximately within the range of 9-12 cluster size, and it becomes almost constant at the end. Therefore, it is better to select the cluster size within this range to optimize power consumption.

With the results obtained in the above case, the impact of the phase duration (or session duration) on the power consumption can be derived as indicated in TABLE VI. In this case, the cluster size has been fixed to 10 as it can be approximated as an elbow point in distributions presented in Fig. 11. With that fixed cluster size, the power usage under five different phase durations (when phase duration is changing, the session duration varies automatically) was derived by neglecting the power usage associated with the synchronization sessions (As these sessions occur in a short period in long intervals, its impact is assumed to be negligible). According to the results obtained from that, it is evident that there is no significant impact from the phase duration for the overall power consumption. However, we have identified a higher chance of escaping some users from the cluster with longer sessions. Therefore, it is recommended to keep the

(a) For 1hour Session

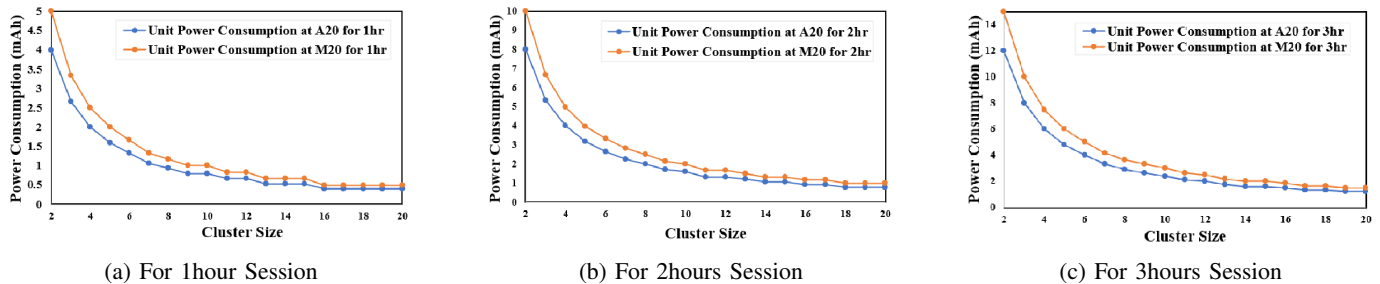(b) For 2hours Session

(c) For 3hours Session

Fig. 11: Impact of Cluster Size for the Application's Power Consumption

phase duration as small as possible (e.g., 1m phase duration, i.e., 10 minutes sessions).

TABLE VI: Power Saving with Clustering Approach Under Different Phase (or Session) Duration - For Samsung Galaxy A20

Phase Duaration (minutes)	1	2	3	4	5
Session Duration (minutes)	10	20	30	40	50
Individual Power Contribution for the Session (mAh)	0.1333	0.2667	0.4000	0.5332	0.6665
Power Consumption without Clustering (mAh) for 10 mobiles	1.333	2.667	4.0000	5.332	6.665
Power Saving Per Session (mAh)	1.1997	2.4003	3.6000	4.7988	5.9985
Power Saving Per Hour (mAh)	7.2				

According to the results obtained in TABLE V, under the regular operation, Samsung Galaxy A20 will use 8mAh power per hour, but with the above approach, it will theoretically be limited to 0.8mAh with 7.2mAh (90%) of total saving.

D. Financial Benefits of Our Solution

With the developed contact tracing system, there are many inherent financial benefits included. For example, a country will continue its economic activities with reduced interruptions due to the COVID-19 pandemic. On the other hand, these automatic contact tracing methodologies save much money, time, and human effort. Especially, as indicated in Fig. 12, governments and medical authorities can identify the required number of PCR tests to be performed by predicting the number of individuals who exceed the predefined threshold limits using the developed model.

As indicated in Fig. 12, we have performed a simulation with 250, 500, 750, and 1000 nodes respectively with a fixed number of connections (6) for each node to identify the behaviors in the population under different periods (10, 20, and 30 days). In addition to that, we considered three probabilities, 0.6, 0.7, and 0.8, to measure the count of these thresholds exceeding users.

Similarly, a medical authority may set a threshold value (variable) for the general public's infection probability by considering the current COVID-19 spread of the country. Moreover, it has to identify the targeted population to be subjected to immediate PCR testing or self-isolation. With these details, our developed system would be capable of identifying the individuals who have exceeded the intended infection probability threshold (as individual contact details are recorded in the system for up to 21 days). With this

approach, a government can quickly identify the required number of PCR tests for a specific population and avoid unnecessary costs by performing tests for the whole population arbitrarily and inefficiently. As it is evident from Fig. 12, the expected reduction of required PCR tests (or self-isolation requirements) from the developed algorithms is almost 15% - 25%. Under each case (Without the algorithm, the whole selected population (250, 500, 750, or 1000) may get subjected to PCR test or self-isolation). Furthermore, they can also identify the testing priorities and perform the PCR tests in a more organized manner throughout the population, improving the efficiency of the whole process. It would also save much time. In addition to that, the medical officers can also grant self-isolation recommendations for individuals according to the requirements.

E. Impact of Energy Saving on Accuracy

As mentioned in Section IV-B, we identified an effect on the system's accuracy with energy-saving. Therefore, we considered a way to analyze this trade-off by detecting the situation where the maximum error could occur with this proposed algorithm. In this case, there are three points highlighted again. First, as indicated from the TABLE VI, we have detected no significant impact on the energy-saving from the phase duration. Second, as mentioned in Section IV-B, after each phase, a session enables both advertising and scanning mode in each mobile in the cluster. Therefore, for an external user to move without being detected by a clustered mobile (which is idle and inactive at the moment), the contact duration should be less than the period of one phase. Here, if we refer to Fig. 13, it is clear that the infection probability increases in a logarithmic-like behavior when increasing the contact period. Therefore, keeping the undetected period minimum between the inactive mobile in the cluster and the out-of-cluster (dynamic) user is better to minimize the probability calculation error. Therefore, we suggest 1 minute for the phase duration due to its very low infection probability within such a short period. According to the organization's requirements, this value can be set up as there is no impact on the cluster's overall energy consumption.

To evaluate this further, we can consider the situation where the maximum error could occur with the developed algorithm. In this case, we can recognize two significant factors. Those factors are the contact period and contact distance. Moreover, under each case, we can consider the situations associated

(a) 250 Nodes with 6 Connections (max)

(b) 500 Nodes with 6 Connections (max)

(c) 750 Nodes with 6 Connections (max)

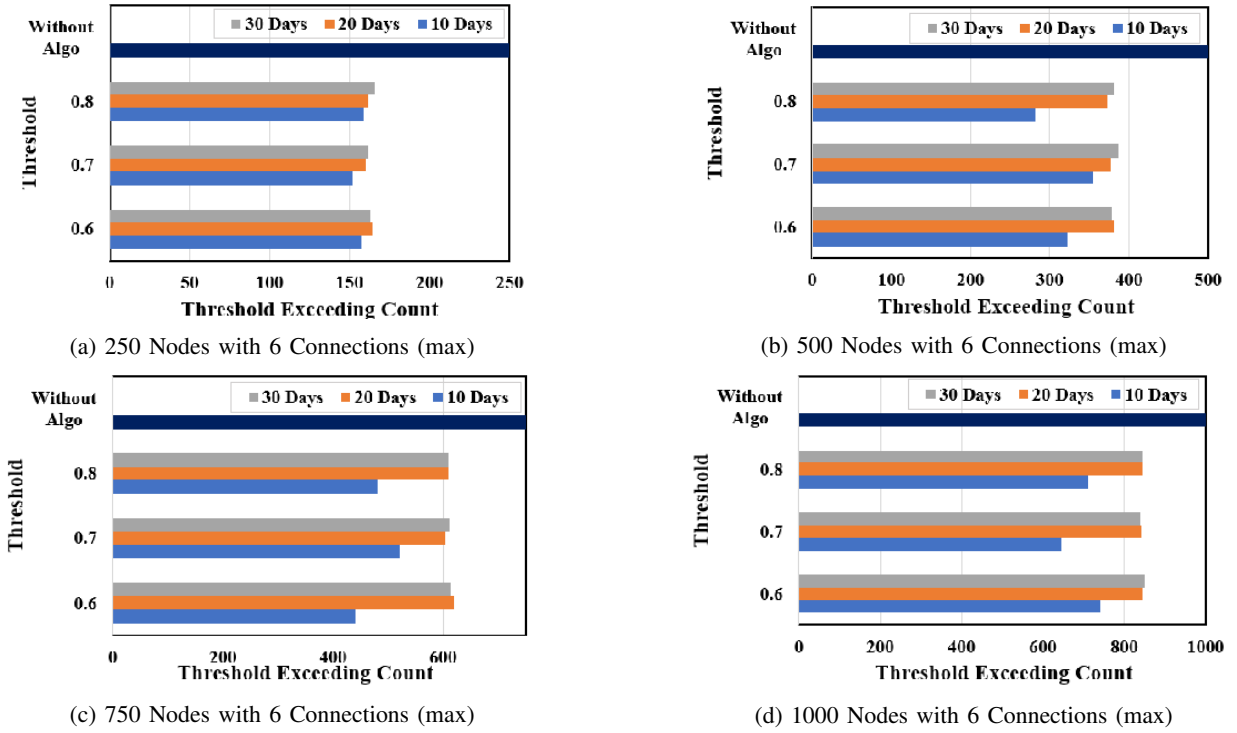
(d) 1000 Nodes with 6 Connections (max)

Fig. 12: Threshold Exceeding Counts Under 250, 500, 750 and 1000 Nodes with 6 Maximum Possible Connections for Each
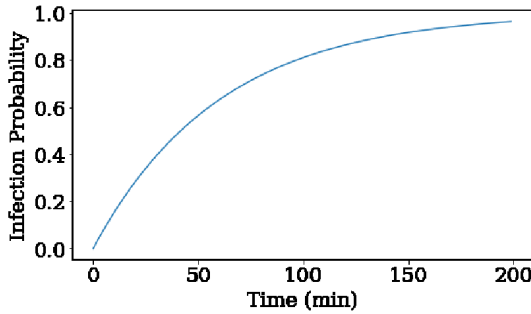

Fig. 13: Change in Probability of Getting Infection with Time of Exposure

with clustered users and out-of-cluster users. The delegated mobile is always looking for incoming advertisements, and there is a minimum chance to miscount any clustered or out-of-cluster mobile contact period. However, there can be an issue associated with the distance calculation between the clustered and out-of-cluster mobiles.

Therefore, according to the worst-case calculation presented with Section IV-B, it is clear that the distance between two users could be calculated as 1 meter when the space is more significant than that. However, the worst distance would be 1 meter as the infection probability gets lower with increasing distance. With this sense, in the following example (which is related to the previous model in Section IV-B), we present the maximum error that could occur with the existing system.

E.g.

External User - a

Broadcasting User - A

Inactive mobile in the cluster - C

Calculated distance from a to A = 8m

Distance from A to C = 8m

Therefore, the minimum distance between a to C = 0m

However, this distance can also be 16m. If the distance is 0m and the contact period is 1min,

from Equation 2,

$$P_d(0m) = 1 \tag{7}$$

from Equation 3,

$$P_t(1min) = 0.016 \tag{8}$$

If we consider infection probability of a as 1, then the infection probability of COVID-19 to C from a, can be calculated from Equation 4. It is calculated without the fixed risk factors.

$$P_i(C, a) = 0.016 \tag{9}$$

On the other hand, if the actual distance between the users is 16m,

from Equation 2,

$$P_d(16m) = 6.25 \times 10^{-6} \tag{10}$$

Therefore, the total probability of C getting the virus infection from a can be calculated again from Equation 4 as,

$$P_i(C, a) = 1 \times 10^{-7} \tag{11}$$

Hence, the worst-case probability loss is 0.016 which is a low contribution. If a stays near the cluster for more than

1min, the exact distance can be calculated, and the error is removed. In this sense, we can conclude that the system error occurring via the presented algorithm is very low, even with the worst-case scenario.

VII. CONCLUSION AND FUTURE WORKS

With this work, we present a contact tracing system that is equipped with BLE and GPS capabilities to address significant drawbacks and concerns highlighted in current COVID-19 patient contact tracing systems. Additionally, this system supports offline and online operating modes, giving relevant authorities and users a high level of flexibility. Besides that, with the aid of a developed algorithm, we could also evaluate and produce infection probabilities for users. Also, we simulated the algorithms with a graph-based approach and compared the findings with actual data from different regions. The algorithms enable generating required alerts to indicate the critical incidences (i.e., close contact with a COVID-19 patient) and push the necessary notifications to users about their associated risk levels in a more tailored fashion than other systems. We also identified that the proposing clustering-based system would provide up to 90% of energy-saving for idle users while providing a 15% - 25% reduction in the required number of Polymerase Chain Reaction(PCR) tests (or self-isolation cases). In that sense, our work offers a higher number of options and accuracy over many other solutions currently being implemented worldwide. Furthermore, the simulation findings assisted in gaining insight into how the system will perform in real-world and practical situations.

We intend to increase the system's user privacy while collecting data in the future by incorporating strong anonymity and unlinkability properties. Furthermore, we want to improve prediction accuracy by combining machine learning-based infection probability predictions with the established approach. Moreover, this system should be helpful and effectively perform in any other pandemic similar to COVID-19, which can spring out any time or anywhere in the future.

REFERENCES

[1] "Naming the coronavirus disease (covid-19) and the virus that causes it." [Online]. Available: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-(covid-2019)-and-the-virus-that-causes-it

[2] S. Begley, Jim, Alan, and C. M, "New Digital Tools Could Speed Up Covid-19 Contact Tracing," Apr 2020. [Online]. Available: https://www.statnews.com/2020/04/02/coronavirus-spreads-too-fast-for-contact-tracing-digital-tools-could-help

[3] A. Greenberg, "How Apple and Google Are Enabling Covid-19 Bluetooth Contact-Tracing." [Online]. Available: https://www.wired.com/story/apple-google-bluetooth-contact-tracing-covid-19/

[4] K. Foy and L. Laboratory, "Bluetooth Signals from Your Smartphone Could Automate Covid-19 Contact Tracing While Preserving Privacy," Apr 2020. [Online]. Available: http://news.mit.edu/2020/bluetooth-covid-19-contact-tracing-0409

[5] "Help Speed Up Contact Tracing with Trace-Together." [Online]. Available: https://www.gov.sg/article/help-speed-up-contact-tracing-with-tracetogether

[6] A. G. D. of Health, "COVIDSafe app," May 2020. [Online]. Available: https://www.health.gov.au/resources/apps-and-tools/covidsafe-app

[7] "Modes of transmission of virus causing COVID-19: implications for IPC precaution recommendations." [Online]. Available: https://www.who.int/news-room/commentaries/detail/modes-of-transmission-of-virus-causing-covid-19-implications-for-ipc-precaution-reco

[8] S. A. Lauer, K. H. Grantz, Q. Bi, F. K. Jones, Q. Zheng, H. R. Meredith, A. S. Azman, N. G. Reich, and J. Lessler, "The Incubation Period of Coronavirus Disease 2019 (COVID-19) from Publicly Reported Confirmed Cases: Estimation and Application," *Annals of internal medicine*, 2020.

[9] K. Townsend, C. Cufí, R. Davidson *et al.*, *Getting started with Bluetooth low energy: tools and techniques for low-power networking.* " O'Reilly Media, Inc.", 2014.

[10] S. Raza, P. Misra, Z. He, and T. Voigt, "Building the Internet of Things with Bluetooth Smart," *Ad Hoc Networks*, vol. 57, pp. 19–31, 2017.

[11] K. Heurtefeux and F. Valois, "Is RSSI a Good Choice for Localization in Wireless Sensor Network?" in *2012 IEEE 26th international conference on advanced information networking and applications*. IEEE, 2012, pp. 732–739.

[12] M. M. Siwiak, P. Szczesny, and M. P. Siwiak, "From a single host to global spread. the global mobility based modelling of the covid-19 pandemic implies higher infection and lower detection rates than current estimates," *The Global Mobility Based Modelling of the COVID-19 Pandemic Implies Higher Infection and Lower Detection Rates than Current Estimates (3/23/2020)*, 2020.

[13] "Aarogya Setu Mobile App." [Online]. Available: https://www.mygov.in/aarogya-setu-app/

[14] P. Hubaux, "Decentralized privacy-preserving proximity tracing," Ph.D. dissertation, Fraunhofer HHI, 2020.

[15] J. Chan, S. Gollakota, E. Horvitz, J. Jaeger, S. Kakade, T. Kohno, J. Langford, J. Larson, S. Singanamalla, J. Sunshine *et al.*, "Pact: Privacy sensitive protocols and mechanisms for mobile contact tracing," *arXiv preprint arXiv:2004.03544*, 2020.

[16] T. Chowdhury, M. Rahman, S.-A. Parvez, A. Alam, A. Basher, A. Alam, and S. Rizwan, "A Multi-step Approach for RSSi-based Distance Estimation Using Smartphones," in *2015 International Conference on Networking Systems and Security (NSysS)*. IEEE, 2015, pp. 1–5.

[17] [Online]. Available: https://www.covid.is/app/en

[18] "App for tracking coronavirus contacts trialled in Finland," May 2020. [Online]. Available: https://www.goodnewsfinland.com/app-for-tracking-coronavirus-contacts-trialled-in-finland/

[19] "Ketju warns of the risk of contagion and alerts you about potential exposure." [Online]. Available: https://ketjusovellus.fi/en/

[20] P. H. O'Neill, "A flood of coronavirus apps are tracking us. Now it's time to keep track of them." Jun 2020. [Online]. Available: https://www.technologyreview.com/2020/05/07/1000961/\\launching-mittr-covid-tracing-tracker/

[21] info@covid19cz.cz, "eRouška startuje." [Online]. Available: https://covid19cz.cz/aktuality/2020-04-14-erouska-startuje

[22] "Proximity tracing and the coronavirus pandemic: Fraunhofer's proposal for a German approach," Apr 2020. [Online]. Available: https://www.fraunhofer.de/en/press/research-news/2020/april/fraunhofer-develops-anti-sars-cov-2-app.html

[23] C. O'Brien, "HSE Covid-19 tracing app data will be stored on individual devices," Apr 2020. [Online]. Available: https://www.irishtimes.com/business/technology/hse-covid-19-tracing-app-data-will-be-stored-on-individual-devices-1.4240304

[24] "ViruSafe-Prevent and assist in the fight against COVID-19 ." [Online]. Available: https://virusafe.info/

[25] "HaMagen - The Ministry of Health App for Fighting the Spread of Coronavirus." [Online]. Available: https://govextra.gov.il/ministry-of-health/hamagen-app/download-en/

[26] [Online]. Available: https://flo.uri.sh/visualisation/2241702/\\embed?auto=1

[27] "MyTrace, a Preventive Counter Measure and Contact Tracing Application for COVID-19," May 2020. [Online]. Available: https://www.mosti.gov.my/web/en/mytrace/

[28] "HOW DOES StopKorona! work?" [Online]. Available: https://stop.koronavirus.gov.mk/en

[29] "What is CovidRadar?" [Online]. Available: http://covidradar.mx/

[30] "helsenorge.no." [Online]. Available: https://helsenorge.no/coronavirus/smittestopp

[31] "Italy launches Immuni contact-tracing app: Here's what you need to know." [Online]. Available: https://www.thelocal.it/20200605/italy-to-begin-testing-immuni-contact-tracing-app-in-four-regions

[32] "TraceTogether, safer together." [Online]. Available: https://www.tracetogether.gov.sg/

[33] "NHS COVID-19 App." [Online]. Available: https://www.nhsx.nhs.uk/covid-19-response/nhs-covid-19-app/

[34] "Download TraceCovid App Now!" [Online]. Available: https://tracecovid.ae/

[35] Webmedia, "News Details." [Online]. Available: https://portal.www.gov.qa/wps/portal/media-center/news/news-details/todeterminecoronavirustransmission\\chainsehterazappisnowavailablefordownload

[36] "Życie po kwarantannie – przetestuj ProteGO - Ministerstwo Cyfryzacji - Portal Gov.pl," Apr 2020. [Online]. Available: https://www.gov.pl/web/cyfryzacja/zycie-po-kwarantannie--przetestuj-protego

[37] D. F. Llorca, R. Quintero, I. Parra, and M. Sotelo, "Recognizing Individuals in Groups in Outdoor Environments Combining Stereo Vision, RFID and BLE," *Cluster Computing*, vol. 20, no. 1, pp. 769–779, 2017.

[38] S. Kaddouri, M. El Hajj, G. Zaharia, and G. El Zein, "Indoor path loss measurements and modeling in an open-space office at 2.4 ghz and 5.8 ghz in the presence of people," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2018, pp. 1–7.

[39] P. Bahl, C. Doolan, C. de Silva, A. A. Chughtai, L. Bourouiba, and C. R. MacIntyre, "Airborne or droplet precautions for health workers treating COVID-19?" *The Journal of infectious diseases*, 2020.

[40] N. van Doremalen, T. Bushmaker, D. H. Morris, M. G. Holbrook, A. Gamble, B. N. Williamson, A. Tamin, J. L. Harcourt, N. J. Thornburg, S. I. Gerber, J. O. Lloyd-Smith, E. de Wit, and V. J. Munster, "Aerosol and Surface Stability of SARS-CoV-2 as Compared with SARS-CoV-1," *New England Journal of Medicine*, vol. 382, no. 16, pp. 1564–1567, 2020. [Online]. Available: https://doi.org/10.1056/NEJMc2004973

[41] V. Surveillances, "The Epidemiological Characteristics of an Outbreak of 2019 Novel Coronavirus Diseases (COVID-19)—China, 2020," *China CDC Weekly*, vol. 2, no. 8, pp. 113–122, 2020.

[42] "Novel Coronavirus (COVID-19) Cases Data." [Online]. Available: https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases

[43] A. J. Kucharski, T. W. Russell, C. Diamond, Y. Liu, J. Edmunds, S. Funk, R. M. Eggo, F. Sun, M. Jit, J. D. Munday *et al.*, "Early dynamics of transmission and control of COVID-19: a mathematical modelling study," *The lancet infectious diseases*, 2020.

Chamara Sandeepa is a Ph.D. student at the School of Computer Science, University College Dublin, Ireland. Currently, he is working in the field of explainable AI for privacy. He graduated in Bachelor of the Science in Electrical and Information Engineering, with a Second Class Honours Upper Division from the University of Ruhuna, Sri Lanka, in 2020. His degree program comprised Electrical, Software Engineering, Telecommunication, Electronics, where he specialized under the Software Engineering sector. He has industry experience as a Software Engineer, and he also worked in the IoT and AI fields. His research interests include AI privacy, IoT, Computer Science, Data Science, and e-Healthcare.

Nadeeka Dissanayaka is currently working as a Access Network Planning Engineer and she has obtained her Bachelor of the Science of Engineering degree specialized in Electrical and Information Engineering (B.Sc.Eng.(Second Class Honours upper Division)),from the department of Electrical and Information Engineering Faculty of Engineering, University of Ruhuna, Galle, Sri Lanka in 2020. She has followed electrical engineering, telecommunication engineering, electronic engineering and software engineering. Her research interests are, IoT, e-Healthcare and electrical engineering aspects.

Charuka Moremada is currently reading for Master of Science in Applied Sciences and Engineering: Applied Computer Science in Vrije Universiteit Brussel, Belgium. He has obtained his, Bachelor of the Science of Engineering degree from the Department of Electrical and Information Engineering, Faculty of Engineering, University of Ruhuna, Sri Lanka in 2020, specialized in Electrical and Information Engineering (B.Sc.Eng.(First Class Honours)). He has co-authored some publications in certain research areas and his current research interests are, Blockchains, Internet of Things, Ambient Assisted Living, BLE-Contact Tracing, Mobile Communications, Artificial Intelligence, machine learning, e-healthcare and bio-medical engineering.

.

Tharindu Gamage is currently working as a Lecturer in the Department of Electrical and Information Engineering, University of Ruhuna, Sri Lanka. He obtained his first degree from the Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka in 2009. He worked as a research assistant for several years in the Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka. During the time he was working as a research assistant, he obtained his M.Sc. from the same department of University of Moratuwa, Sri Lanka in 2014. His research interests are IoT, Embedded Systems, High Performance Computing and Medical Image Processing. URL: http://eie.eng.ruh.ac.lk/team/tharindu-gamage/

Madhusanka Liyanage (Senior Member, IEEE) received his B.Sc. degree (First Class Honours) in electronics and telecommunication engineering from the University of Moratuwa, Moratuwa, Sri Lanka, in 2009, the M.Eng. degree from the Asian Institute of Technology, Bangkok, Thailand, in 2011, the M.Sc. degree from the University of Nice Sophia Antipolis, Nice, France, in 2011, and the Doctor of Technology degree in communication engineering from the University of Oulu, Oulu, Finland, in 2016. From 2011 to 2012, he worked as a Research Scientist at the I3S Laboratory and Inria, Sophia Antipolis, France. He is currently an assistant professor/Ad Astra Fellow at the School of Computer Science, University College Dublin, Ireland. He is also acting as an adjunct Processor at the Center for Wireless Communications, University of Oulu, Finland and Director of Graduate Research at the School of Computer Science, University College Dublin, Ireland. He was also a recipient of the prestigious Marie Skłodowska-Curie Actions Individual Fellowship during 2018-2020. During 2015-2018, he has been a Visiting Research Fellow at the CSIRO, Australia, the Infolabs21, Lancaster University, U.K., Computer Science and Engineering, The University of New South Wales, Australia, School of IT, University of Sydney, Australia, LIP6, Sorbonne University, France and Computer Science and Engineering, The University of Oxford, U.K. He is also a senior member of IEEE. In 2020, he received the "2020 IEEE ComSoc Outstanding Young Researcher" award by IEEE ComSoc EMEA. Dr. Liyanage is an expert consultant at European Union Agency for Cybersecurity (ENISA). In 2021, Liyanage elevated as Funded Investigator of Science Foundation Ireland CONNECT Research Centre, Ireland. Moreover, he is an expert reviewer at different funding agencies in France, Qatar, UAE, Sri Lanka, and Kazakhstan. Dr. Liyanage's research interests are 5G/6G, SDN, IoT, Blockchain, MEC, mobile, and virtual network security. More info: www.madhusanka.com