

# Federating Digital Contact Tracing using Structured Overlay Networks

Silvia Ghilezan<sup>1,2</sup>, Simona Kašterović<sup>2</sup>, Luigi Liquori<sup>4</sup>, Bojan Marinković<sup>3,1</sup>, Zoran Ognjanović<sup>1</sup>, Tamara Stefanović<sup>2</sup>

<sup>1</sup> Mathematical Institute of the Serbian Academy of Sciences and Arts  
Belgrade, Serbia

{bojanm, zorano}@mi.sanu.ac.rs

<sup>2</sup> Faculty of Technical Sciences, University of Novi Sad,  
Novi Sad, Serbia

{gsilvia, simona.k, tstefanovic}@uns.ac.rs

<sup>3</sup> Clarivate, Serbia

<sup>4</sup> INRIA & Université Côte d'Azur, France  
Luigi.Liquori@inria.fr

**Abstract.** In this paper we present a comprehensive, yet simple, extension to the existing systems used for Digital Contact Tracing in Covid-19 pandemic. The extension, called *BubbleAntiCovid19 (BAC19)*, enables those systems, regardless of their underlying protocol, to enhance their sets of traced contacts and to improve the global fight against pandemic during the phase of opening borders and enabling more traveling. *BAC19* is a structured overlay network, or better, a Federation of mathematical Distributed Hash Tables. Its model is inspired by the Chord and Synapse structured overlay networks. The paper presents the architecture of the Overlay Network Federation and shows that the federation can be used as a formal model of Forward Contact Tracing.

**Keywords:** Covid-19, Digital Contact Tracing, Distributed Hash Tables, Structured Overlay Networks, Bluetooth, GPS.

## 1. Introduction

One of the biggest challenges of today is to slow down the spreading of SARS-CoV-2 virus producing Covid-19 pandemic; *Prevention, Testing and Tracing* are the main pillars of the solution. Contact Tracing of an infected person is essential to control the spread of the disease.

*Tracing.* Contact tracing is the process of identifying, notifying, and monitoring people who came in close contact with an individual who was tested positive for an infectious disease, like Covid-19, while he/she was infectious. Contact tracing benefits the fight with the pandemic at multiple levels. Identifying and quarantining close contacts limits their ability to spread the disease. Therefore, in a period in which the disease and its effects are still being investigated, contact tracing plays a key role in preventing the further spread of the disease. Furthermore, contact tracing data helps medical experts to find the origin of the virus and learn more about the nature of the virus.

*Manual Contact Tracing.* Contact tracing has mostly been done manually since many centuries ago just by taking note on a simple piece of paper the list of persons and goods

1 you get in contacts with (see e.g. *La Peste* by A. Camus [6]). In the actual days, manual  
 2 contact tracing could be exploited using simple telephone calls. Identifying contacts is  
 3 done through an interview with the person infected with the virus. Each person is then  
 4 contacted by phone. Health Authorities should quickly alert people who are close contacts  
 5 that they may have been exposed to the virus. The sooner the contacts are notified, the  
 6 lower the risk of the spreading further. However, due to the highly contagious nature of  
 7 the SARS-CoV-2 virus and the fact that symptoms can manifest after many days (or even  
 8 never, e.g. *asymptomatic cases*), manual contact tracing does not give satisfactory results.  
 9 Health departments and authorities do not have enough employees to do manual contact  
 10 tracing. It must be further emphasized that the SARS-CoV-2 can be transmitted not only  
 11 by direct contact, but also by indirect contact. The reason is that infected people can leave  
 12 virus droplets on any physical object they touch. In this case, manual contact tracing  
 13 is ineffective. For the reasons stated above, digital contact tracing has been considered  
 14 already at the beginning of the Covid-19 pandemic.

### 15 **1.1. Problem**

16 There is a plethora of digital contract tracing applications in use all over the world fighting  
 17 the Covid-19 pandemic [9, 19]. They are developed on very different paradigms, central-  
 18 ized [7] vs. decentralized [28], GPS based (very few indeed because of a clear violation  
 19 of privacy) vs. Bluetooth Low Energy based (the majority). The rush to make these ap-  
 20 plications work in the shortest time led to their great diversity. The most important open  
 21 problem is their interoperability. There are many ongoing efforts to make a federation of  
 22 these different systems. Herein, we address this problem and propose a solution based on  
 23 mathematical models of overlay networks.

### 24 **1.2. Contributions**

25 We develop a formal federation overlay network, called *BubbleAntiCovid19 (BAC19)*,  
 26 for connecting different digital contact tracing applications, which are currently in use  
 27 all over the world. The model is based on the well-known model of Structured Overlay  
 28 Network protocols like e.g. Chord [24, 25], Kademia [20], Synapse [14]. We prove that  
 29 *BAC19* provides a complete and fully exhaustive retrieving procedure of people that get  
 30 in touch with other people having tested positive to the Covid-19 disease. Hence, *BAC19*  
 31 is proven to be a simple yet powerful *interconnection* of already existing digital contact  
 32 tracing applications that - by construction - do not communicate with each others as such  
 33 providing their efficient interoperability.

34 As far as we know, the mathematical model and techniques presented in this paper  
 35 have not been considered in other approaches.

### 36 **1.3. Overlay networks in a Nutshell.**

37 Structured Overlay Networks are suitable models of scalable and efficient organization of  
 38 resources on the Internet. They represent logical organizations, independent on underlying  
 39 network infrastructure that physically connects available assets. Overlay networks have  
 40 been proven as very resilient tool in the situation when some parts of the underlying  
 41 infrastructure fail or become overloaded or corrupted.

#### 1 **1.4. Organization of the Paper**

2 The rest of the paper is organized as follows. Section 2 presents classifications of digital  
3 contact tracing applications. Section 3 reviews Chord and Synapse protocols of overlay  
4 networks. Section 4 briefly reviews basic notions of Abstract State Machines and some  
5 related work by the authors. Section 5 introduces the *BAC19* system and proves the com-  
6 pleteness and full exhaustiveness of the retrieving procedure. Section 6 presents a discus-  
7 sion on other proposals for providing interoperable frameworks for digital contact tracing.  
8 Section 7 concludes the paper. Appendix give an overview of different digital contact trac-  
9 ing applications that are in current use against the pandemic.

## 10 **2. Digital Contact Tracing Applications**

11 Advances in digital technology have enabled smartphones and other digital devices to  
12 be used for contract tracing. Particularly, more and more countries are showing interest in  
13 digital contact tracing applications (DCT apps) implemented for smartphones. Despite the  
14 great variety among these applications, all contact tracing apps work on the principle of  
15 automatic data exchange with nearby devices. When a user of a particular contact tracing  
16 app is identified as infected, a special report is uploaded to the DCT app server. Based on  
17 that report, close contacts of the infected person (also DCT app users) are informed that  
18 they have been in a contact with a positive user and/or the app calculates their exposure  
19 risk. The identity of the infected persons is not disclosed in order to protect their privacy.  
20 Existing contact tracing apps can be classified based on two criteria:

- 21 – Contact-tracing technology;
- 22 – System architecture.

23 More information about the classification of the existing contact tracing apps can be found  
24 in [23,26].

### 25 **2.1. Classification by Contact-Tracing Technology**

26 In order for two people to be in close contact, they need to stay in the same place, at a  
27 short distance, for a long enough period of time. Therefore, the main data type used by  
28 the contact tracing apps is location data. There are various technologies for collecting and  
29 tracking location data, and contact tracing apps can be divided into two major categories  
30 depending on whether they track absolute or relative location:

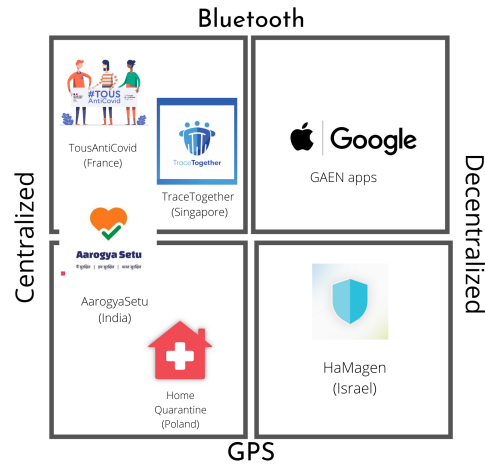
- 31 – Absolute location apps – Apps that track the absolute location of their users are  
32 mostly based on GPS technology. Location data is stored in the form of geolocation  
33 coordinate pair. These apps are also known as Geolocation-based DCT apps.
- 34 – Relative location apps – Apps that track relative location of their users are mostly  
35 based on Bluetooth technology. These apps are also known as proximity DCT apps.  
36 A boarding pass or a ticket for a specific event can also be considered as relative  
37 location data. In order to use this kind of data, some contact tracing apps deploy QR  
38 code technology.

## 2.2. Classification by System Architecture

Since the data is collected from users, their processing should be addressed. When it comes to DCT app data managing, the responsibility can be on a central authority or on each user individually. Therefore, contact tracing apps can be divided into three major categories depending on the architecture of the underlying system:

- Centralized apps - data are solely managed by a central server;
- Hybrid apps - multiple nodes can manage the data, but the control is centralized;
- Decentralized apps - each user is managing his/her data.

In centralized apps, the central server is responsible for ID generation, risk analysis and notifications. In decentralized apps these functionalities are moved to the user devices and the central server is only an encounter point. The hybrid architecture proposes decentralized ID generation and centralized risk analysis. There are a few proposed hybrid contact tracing protocols, but their implementation in real contact tracing apps is still waiting. More about these protocols can be found in [2]. For that reason, we will focus on apps with centralized and decentralized architecture, and we will provide an overview of the existing contact tracing apps based on the above classifications in Appendix. A. The results are summarized in Figure 1, which is motivated by [23].



**Fig. 1.** Classification of analyzed applications

## 2.3. Geolocation-based Apps.

Geolocation-based DCT apps record past geo-trajectories of every user, and the calculation of exposure risk of a user is based on the intersection of its past trajectories and trajectories of patients. We give a brief review of the existing Geolocation-based apps in Appendix A.1.

Two main advantages of geolocation-based DCT apps are the following:

- 1) Geolocation-based DCT apps are compatible with manual contact tracing. Compatibility of geolocation-based DCT apps and manual contact tracing has mutual benefits. On the one hand, past geo-trajectories of a patient can be added to an app by the contact tracer even if the patient did not use the app. This enables the app to warn more users. On the other hand, the app can give the information about places with higher exposure risk to a contact tracer, so that the contact tracer can identify high-risk service workers.
- 2) Another advantage is that geolocation-based DCT apps can recognize patterns of disease's spreading and locations with higher exposure risk, and they can inform health authorities about it.

Nevertheless, geolocation-based DCT apps have also disadvantages. The major challenges are privacy concerns, which cause low adoption rate of these applications. User's privacy can be violated in several ways. Recording all user's trajectories can result in revealing user's personal information such as identity, home address, work address, the identity of the patient and revealing user's exposure risk to other users. These problems have been elaborated in more details in [8].

#### 2.4. Bluetooth-based Apps

Bluetooth-based DCT apps record direct contacts of the users. A device generates a unique, randomized identifier and assigns it to a user. There are two kinds of identifiers: static, identifiers do not change over time, and dynamic, identifiers change over time. During a direct contact devices exchange identifiers and save received identifiers. Once a user is identified as positive in the application, other users can calculate their exposure risk by checking whether they received a patient's identifier. We give a brief review of the existing Bluetooth-based apps in Appendix A.2.

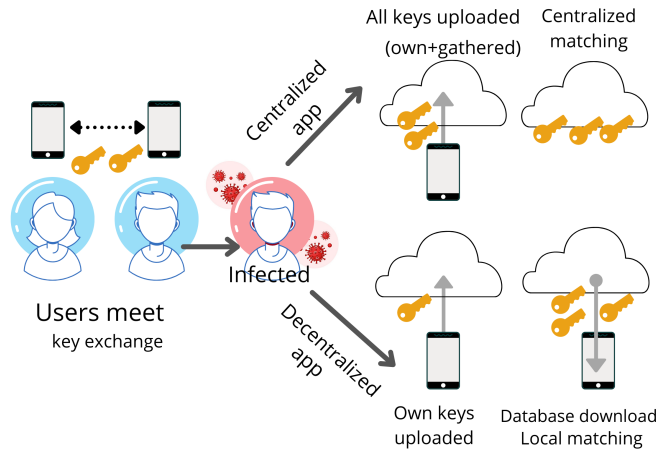
Depending on whether the exposure risk is calculated by the central server or the user's device, we have centralized and decentralized apps, respectively, see Figure 2, which is motivated by [13].

Centralized apps raise privacy concerns and questions about massive surveillance. People often do not trust servers and as a consequence there is a low adoption rate of these applications. On the other hand, the advantage of centralized apps is the possibility for health authority to make a transmission graph and learn more about the virus. Also, the possibility of false positive users is reduced.

As we have already observed, privacy issues lead to low adoption rate and decrease the efficiency of the application. In order to solve the problem of distrust of the central server, decentralized apps were designed. However, decentralized methods also raise some privacy concerns, for example the identity of a patient can be easily revealed.

The major disadvantage is that Bluetooth-based DCT apps work only if both users have installed the *same* application.

In order to take advantage of both types of these apps, Bluetooth-GPS apps were designed, see Appendix A.3. Given the different characteristics of DCT apps, the question arises whether it is possible to aggregate their data in order to track contacts more effectively. The answer can be found in overlay networks.



**Fig. 2.** Centralized vs Decentralized Bluetooth-based apps

### 1 3. Overlay Networks

2 Overlay networks are the way to organize available assets, as mentioned in Section 1.

3 Some overlay networks are implemented in a form of Distributed-Hash-Tables (DHTs).  
 4 One of DHT protocols is the Chord protocol. It was introduced in [24,25]. Nodes that are  
 5 part of a Chord system form a ring shaped network. The basic operations of a Chord node  
 6 are entering and leaving the system and the mapping given key onto the corresponding  
 7 node of the system using consistent hashing.

8 The correctness and efficiency of the Chord's protocol lookup procedure was in the  
 9 focus of several papers, e.g. [16, 18, 24, 25]. However, these properties will not be in the  
 10 focus of this paper. Our goal is to deliver information of every affected node, so we will  
 11 not use any presented improvements to speed up the process of getting results, but to  
 12 linearly pass every node in a Chord network, to be sure that no information is missed.

13 Interconnection of several overlay networks is a very hard problem since different  
 14 networks may use different protocols, and even in the case of several DHT networks that  
 15 use the same protocol (e.g. Chord) it is enough that every overlay network uses *its own*  
 16 *hash function* and information between two of them cannot be exchanged. A proposal to  
 17 solve this issue was given by defining the Synapse protocol in [14]. Its performances were  
 18 analyzed in [15], whereas one real-life proof of concept was developed in [17]. For the  
 19 purpose of this paper we will consider the so-called, *white-box* version of the Synapse  
 20 protocol that, in short, allows to consider all the keys as they were using the same hash  
 21 table (see [14] for details). Again, since we will use the linear search procedure in one  
 22 Chord network we can be sure that information will be retrieved if it exists in the system.

### 23 4. Abstract State Machines and Chord

24 In this section, we briefly review basic notions of Abstract State Machines.

1 Abstract State Machine (ASM) [5, 11] is a formalization method to model algorithms  
 2 at the appropriate abstraction level. An ASM  $\mathcal{A}$  is defined as a program  $Prog$  which  
 3 consists of:

- 4 – an at most countable set of states, its subset of initial states, and
- 5 – a finite number of transition rules,

6 where states are first order structures over a fixed signature, whereas transitional rules:

- 7 – update ( $:=$ ),
- 8 – sequential ( $seq \dots endseq$ ),
- 9 – conditional ( $if \dots then \dots else \dots endif$ ),
- 10 – parallel ( $par \dots endpar$ ),
- 11 – nondeterministic ( $choose v \in U \text{ satisfying } g(v) \dots endchoose$ ) and
- 12 – universal ( $forall v \text{ with } g \dots endforall$ )

13 represent next-state functions. An execution of one of the last two types of rules introduces  
 14 a variable  $v$ . In the case of nondeterministic rule, the transition is executed with a value of  
 15  $v$  which satisfies a guard  $g$ , while in the case of universal rule, the transition is executed  
 16 simultaneously for all values  $v$  which satisfy a guard  $g$ . An ASM can interact with its  
 17 environment using external functions (oracles) by providing arguments to oracles and  
 18 receiving the corresponding results.

19 In a distributed case with many agents, every agent executes its own program and  
 20 has its own partial view of a global state. The nullary function  $Me$  allows an agent to  
 21 identify itself among other agents. The global program is the union of all agents' pro-  
 22 grams, whereas a transition between two states is obtained by an evaluation of transition  
 23 functions of all agents.

24 An ASM  $\mathcal{A}$  models a real system  $\mathcal{S}$  in terms of evolution of states described by runs.  
 25 A run of  $\mathcal{A}$  is a (in)finite sequence of  $S_0, S_1, S_2, \dots$  where  $S_0$  is an initial state, and  
 26 every  $S_{i+1}$  is obtained from  $S_i$  by executing a transitional rule. In this paper we consider  
 27 only the runs in which states are global and agents' moves are atomic (instantaneous).  
 28 The most general kind of runs for a distributed ASM are partially ordered runs. To prove  
 29 properties of partially ordered runs, thanks to the results proved in [11], the attention  
 30 may be restricted to their linearizations that are sequential runs and satisfy the following  
 31 fundamental properties:

- 32 – All linearizations of the same finite initial segment of a run have the same final state.
- 33 – A property holds in every reachable state of a run iff it holds in every reachable state  
 34 of any of its linearization.

35 Note that this implies that it is enough to find only one sequence of transitions and the  
 36 runs that are considered here and start from the same initial state will have the same final  
 37 state.

38 The key notions introduced in [16, Definition 5.1] are:

- 39 – stable states in a Chord network, where a state of a network is stable if the successor  
 40 (predecessor) pointers of all nodes form an ordered ring, and
- 41 – regular runs, where a run is regular if it is a linearization such that nodes leave and  
 42 enter the network only in stable states.

1 Having these notions, the paper [16] proves that the presented formalization of Chord  
 2 consistently maintains the topological structure of rings and manipulates with distributes  
 3 keys. In Section 5 we will explain that our model of *BAC19* satisfies the mentioned con-  
 4 straints and that results from [16] hold also for *BAC19*.

5 On the other hand the papers [14, 15] recognize the fact that the search procedure of  
 6 the Synapse protocol is not complete and fully exhaustive. This is due to the fact that the  
 7 lookup procedure of the Chord network can skip some of the synapse nodes, and thus not  
 8 to spread the query to all networks that are reachable. To avoid this situation we redefine  
 9 the lookup procedure with Algorithm 1, not to skip any node.

## 10 5. System *BAC19*

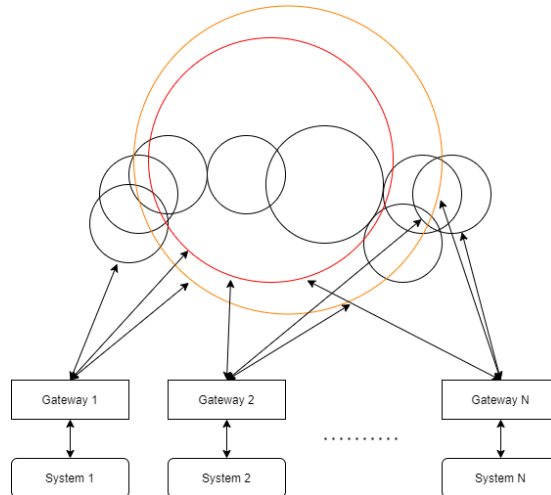
11 In this section, we propose the design of the system *BubbleAntiCovid19* (*BAC19*), which  
 12 is a formal federation overlay network for connecting different digital contact tracing  
 13 applications that are currently in use all over the world.

14 *BAC19* federation (Figure 3) consists of several Chord networks:

- 15 – a network for each person/device of his/her first contacts (black circles in Figure 3),
- 16 – dedicated *red* network to connect all infected persons (red circle in Figure 3),
- 17 – dedicated *amber* network to connect all the first contacts of infected persons (amber  
 18 circle in Figure 3),

19 and

- 20 – *Gateways* (black rectangles in Figure 3) as the connections to the existing digital  
 21 contact tracing systems (black rounded corners rectangles in Figure 3).



**Fig. 3.** *BAC19* Federation Overlay Network



**Algorithm 1:** FindSuccessor

---

```

FINDSUCCESSOR =
For Given key
//successor(id(Me)) is responsible for key
if member_of(key, id(Me), successor(id(Me))) then
| Respond With successor(id(Me))
else
| //Me forwards query to its successor
| Forward Query To successor(id(Me))

```

---

1 The first connection between the proposed extension and an existing system for con-  
2 tact tracing is called *Gateway*. The purpose of a *Gateway* is to maintain communication  
3 between two parts and to transform messages in a way that both sides can communicate  
4 efficiently.

5 The most important thing is to maintain the mappings between identifiers (IDs) used  
6 on both sides of a *Gateway*. As we could see in Section 2, some systems periodically  
7 change IDs, so the possibility to trace those changes is vital for functioning of *BAC19*.  
8 Regarding IDs, our goal is to have one identifier per one person/device regardless of how  
9 many systems it appears in. We argue that this is possible to achieve. First, it is possible  
10 to use sufficiently large codomain of the hash function (e.g.  $2^{128}$ ). Also, it is possible to  
11 select enough parameters of a person/device so that it can be uniquely identified. We are  
12 not storing any other attribute of a person/device except a newly introduced identifier in  
13 our extension.

14 More precisely, with respect to the specifications that are provided in [15, 16] we need  
15 to introduce the following changes:

- the set

$$Network = \{red, amber, net_1, \dots, net_N\}, N \in \mathbb{N}$$

16 to denote all possible networks, where  $N$  is the number of possible persons/devices  
17 in the proposed extension;

- the set *Time* and the function

$$contact\_time() : (Chord \cup \{amber\}) \times Chord \rightarrow Time$$

18 to denote the time of the contact between two persons;

- the external function *current\_date()* to get the current date.

20 To obtain *completeness* and *full exhaustiveness* of the retrieval procedure the rule FIND-  
21 SUCCESSOR, which finds a responsible node for a given ID, is changed as in the way  
22 presented in Algorithm 1.

23 With this proposal we are not compromising performances of the extension by much.  
24 Since the number of contacts of a person is relatively small, it is manageable to allow  
25 increasing the complexity of the worst case retrieval from  $O(\log N)$  to  $O(N)$ . In the  
26 predefined time-slots our extension will receive the following information from a system:

- 27 – all identified infected cases since the last import (Algorithm 2),
- 28 – all confirmed cases that are not infected anymore since the last import (Algorithm 3),

---

**Algorithm 2: Put**

---

```

For all  $inf \in NewCases$ 
  Invoke PUT Of Network  $red$  To Store  $inf$ 
For all  $id \in net_{inf}$ 
  if  $contacttime(amber, id) < t$  or  $contacttime(amber, id) = undef$  then
     $\lfloor$  Set  $contacttime(amber, id) = t$ 

```

---



---

**Algorithm 3: Leave**

---

```

For all  $inf \in Healed$ 
  Invoke LEAVE Of Network  $red$  for  $inf$ 

```

---

- 1 – all identified contacts since the last import in the form of the tuple  $\langle id_i, id_j, t \rangle$  with  
2 meaning that persons  $id_i$  and  $id_j$  had a risk contact at  $t$  timestamp. For the purpose  
3 of providing privacy protection timestamp should be kept at the precision of days.  
4 Unfortunately, this type of communication is not possible with the systems that are  
5 categorized as decentralized Bluetooth systems, since the fact that contact tracing  
6 computation is performed at users' devices and not shared with the central storage.  
7 These systems can only share newly identified cases and their time of recovery (see  
8 Algorithm 4).

---

**Algorithm 4: Set contact time**

---

```

par
  Set  $contacttime(id_i, id_j) = t$ 
  Set  $contacttime(id_j, id_i) = t$ 
endpar

```

---

9 Also, if needed it is possible to introduce the new *Gateway* with the purpose to enter  
10 manually recognized contacts to the system.

11 When information is received from origin systems, as the first step *BAC19* will connect  
12 all newly recognized infected cases to the *red* network, as well as to remove all cured.  
13 A node will remain in the *red* network until its recovering is confirmed. All IDs that are  
14 recognized as the risk contacts of a person/device (e.g.  $id_i$ ) will be added to its bubble.  
15 They will stay there until  $t + 14$  days, where  $t$  is the time of their contact. If the  $id_i$  is the  
16 member of the *red* network all the members of its network will be added to the *amber*  
17 network and stay there during the same time frame  $t + 14$  days. If a contact is already in  
18 the *amber* network timestamp will be updated to the higher value (Algorithm 5).

19 During the opposite way of communication, *BAC19* will pass on information to all  
20 nodes in the *amber* network to *Gateways*. If an identifier is recognized in the set of map-  
21 pings for the particular origin system, the corresponding information is transferred to the  
22 origin system to alert (if not already) the person/device that she/it had risk contact with an  
23 infected person at stored timestamp. Also, *BAC19* is capable to send information on the  
24 second level contacts (the result is stored in the set *Result*, Algorithm 6).

---

**Algorithm 5:** Leave of network

---

```

For all  $net_{id_i} \in Network \setminus \{red\}$ 
  For all  $id_j \in net_{id_i}$ 
    if  $contacttime(id_i, id_j) + 14 \text{ days} > currentdate()$  then
      Invoke LEAVE OF Network  $id_j$  for  $id_i$ 

```

---



---

**Algorithm 6:** Get all nodes

---

```

seq
  Invoke GET all nodes from amber and store the result in Amber
  For all  $id \in Amber$ 
    Invoke GET all nodes from  $net_{id}$  and append the result to Result
endseq

```

---

1 Namely, for all nodes of the *amber* network it is possible to go through every origin  
2 bubble and pass those identifiers to the *Gateways*. Then the origin systems can inform  
3 those persons that they should increase their awareness since they are second level con-  
4 tacts.

5 Using the results from [15, 16] we prove the following statement:

6 **Theorem 1** *The proposed extension stores and retrieves only up-to-date information on*  
7 *Covid-19 positive cases (identified by the origin systems) and their contacts and makes it*  
8 *available to all origin systems.*

9 *Proof.* It is shown in the paper [16] that it might happen that stable states in a Chord  
10 network cannot be achieved if Leave and/or Put rules are fired in unstable states. Thus,  
11 to avoid that in BAC19, it is necessary to ensure that the executions of each of Algorithm  
12 2 - Algorithm 6 do not intertwine.

13 Executions of the proposed extension are performed in the controlled environment.  
14 Due to the scheduled time intervals for running different tasks, the nodes' leaving from the  
15 bubbles will not happen during the unstable states, i.e., there will be only runs compatible  
16 with conditions of [16, Theorems 5.3, 5.4 and 5.7]. Also, the fact that the rule FIND-  
17 SUCCESSOR is changed guarantees that all nodes will be contacted during the search  
18 procedure, and that the retrieving procedure of the Synapse protocol [15] is complete and  
19 fully exhaustive.

20 As a consequence of the mentioned adaptations of the proposed Chord model, all  
21 possible execution of BAC19 fulfill conditions from [16, Theorems 5.3, 5.4 and 5.7]. So,  
22 with these modifications starting from the given state BAC19 will always reach the stable  
23 state, and the retrieved information will be up-to-date and valid.

## 24 6. Discussion

25 The paper [29] proposes building a common API. This approach is rather similar to the  
26 extension proposed in this paper. However, these approaches have also two significant  
27 differences:

- 1 – while [29] building API connection points between each of two different origin sys-
- 2 tems that are connected, our extension proposes a version to common bus where each
- 3 of the origin systems communicates with the proposed extension and in this way re-
- 4 duces and simplifies the number of connection points that needs to be maintained
- 5 when several origin systems are connected;
- 6 – with *BAC19* we are simplifying also information that is being exchanged, and we
- 7 do not violate privacy in the origin systems (since our extension does not collect
- 8 information of origin DCT system).

9 ETSI GS-E4P presents in [10] an interoperability framework for pandemic contact trac-

10 ing systems which allows the centralized and decentralized modes of operation to fully

11 interoperate.

12 A guideline on Interoperability specifications for cross-border transmission chains

13 between approved apps by the European Community [22] proposes a Federation Gate-

14 way Service for synchronizing the diagnosis keys (keys of infected users) across backend

15 servers of each national app. However, this approach focuses only on Google/Apple ex-

16 posure notification apps because the majority of European countries have developed this

17 kind of apps, and also because one Google/Apple exposure notification app can detect the

18 contact with a user of another Google/Apple exposure notification app. In this paper we

19 do not focus on a certain type of DCT apps, we want to achieve the connection between

20 them regardless the contact-tracing technology and their system architecture. We leave to

21 the the reader to envisage the following scenario:

- 22 – Alice lives in the region which has centralized DCT *System A*, while Bob lives in
- 23 the region which has centralized DCT *System B*. Bob has spent some time in the
- 24 region A, and both of them are traveling together side by side with negative RT-PCR
- 25 tests. However, Bob developed symptoms of Covid-19 after couple of days and was
- 26 confirmed as positive.

27 If *System A* and *System B* are part of *BAC19*, it would be enough that only one of Alice

28 and Bob had installed system from the other region just in the time of travel for Alice to

29 be informed that she is the first contact of an infected person.

## 30 7. Conclusions

31 In this paper we have presented *BAC19* a new and efficient overlay network connecting

32 existing systems for digital contact tracing. The advantages of *BAC19* (its usage) are:

- 33 – a person does not install anything new on his/her mobile device (except a new appli-
- 34 cation which is used in the region that this person is visiting);
- 35 – the overlay does not store any personal sensitive information;
- 36 – the overlay is independent regarding how the origin system calculated contacts or is
- 37 it based on Bluetooth or GPS technology;
- 38 – the overlay supports manual entry of recognized contacts;
- 39 – there are no new highly complicated calculations of possible contacts beside those
- 40 that are performed by the original contact tracing systems.

1 The presented extension *BAC19* is the so-called forward tracing system (finding all  
2 contacts of an infected person). We plan to explore the possibilities to adapt *BAC19* to  
3 also enable backward tracing (finding the source of infection using contacts).

4 **Acknowledgment.** This work was partly supported by: the Science Fund Republic of  
5 Serbia #6526707 AI4TrustBC.

## 6 **A. DCT apps - overview**

### 7 **A.1. Geolocation-based DCT apps**

8 *Home Quarantine.* At the beginning of the Covid-19 pandemic, Ministry of Digital Affairs  
9 of Poland developed the Home Quarantine app. More details about this app can be found  
10 in [27]. This is a typical example of a centralized app which deploys GPS technology.  
11 It is developed to support the authorities, especially the police and social services, with  
12 adequate information about people undergoing mandatory home quarantine. Users are  
13 also required to upload their digital photos. So, aside the GPS technology the app also  
14 uses face recognition. The app is mandatory for anyone who has developed coronavirus  
15 symptoms. It should be emphasized that Poland also developed the ProteGO Safe app  
16 for alerting users of close contact with an infected person based on The (Google/Apple)  
17 Exposure Notification (GAEN) system.

18 *The Shield (HaMagen).* In March 2020, Israeli Ministry of Health developed The  
19 Shield app [3]. This is a typical example of a decentralized app which deploys GPS tech-  
20 nology. Location data is stored in the phone. If a user tests positive, he/she can upload  
21 his/her location history to the central server. Once the user uploaded his/her location his-  
22 tory, it is added into a JSON file that is updated with new data on an hourly basis. Matching  
23 the locations happens on the phone. If the match is found, the app shows you the exact  
24 time and location. The app is later updated to work with Bluetooth technology but on a  
25 voluntary basis, every user can choose whether to use the proximity data or not.

### 26 **A.2. Bluetooth-based DCT apps**

27 *Blue-Trace protocol apps.* Singapore's Government Technology Agency in collaboration  
28 with Ministry of Health in March 2020 released the TraceTogether app that allows digital  
29 contact tracing using the custom BlueTrace protocol. Australia has later adopted the pro-  
30 tocol and released the CovidSafe app. More details about these apps can be found in [1].  
31 Contact tracing is done using Bluetooth Low Energy and proximity data is encrypted and  
32 stored only on the users phone. Users in the contact log are identified using anonymous  
33 time-shifting "temporary IDs". If a user tests positive for the infection, the Ministry of  
34 Health requests his/her contact log. The user has the right to choose whether to share the  
35 contact log or not. If the user chooses to share the log, the contact log is uploaded to a  
36 central server and the health authority is then responsible for matching the log to con-  
37 tact detail and informing close contacts of the infected user. These apps are examples of  
38 Bluetooth-based centralized apps. It should also be noted that Singapore solved the prob-  
39 lem of tracing people who don't use smartphones by enabling the app to work with Token  
40 - a physical Bluetooth-based device.

41 *ROBERT protocol app.* The French National Assembly released the StopCovid app  
42 in May 2020 [21]. The app has later been renamed to TousAntiCovid. It allows digital

1 contact tracing using the ROBust and privacy-presERving proximity Tracing protocol  
 2 (ROBERT protocol). It also deploys Bluetooth technology and belongs to the category  
 3 of centralized apps. The difference between this app and apps based on the BlueTrace  
 4 protocol relates to confirmation of positive users. More precisely, in France when a person  
 5 is confirmed to be positive, the lab gives a patient a QR code and the scanned code is the  
 6 proof for the app that you are infected. It is up to you to share this information with  
 7 the app, and if you choose to share this information with a central server, the server is  
 8 responsible for alerting your close contacts.

9 *Google/Apple exposure notification apps.* In April 2020 Google and Apple announced  
 10 the joint work on decentralized Bluetooth-based protocol named The (Google/Apple) Ex-  
 11 posure Notification (GAEN) system [12]. Many states then developed different apps us-  
 12 ing the Google/Apple Exposure Notification framework including Austria (Stopp Corona  
 13 app), Germany (Corona-Warn-App), Italy (Immuni), Canada (COVID Alert ) etc. The  
 14 principle by which applications work is as follows. During a close contact, user’s phones  
 15 exchange random Bluetooth identifiers. These identifiers change frequently and the infor-  
 16 mation about exchanged ID’s is stored on the user’s phone. When a user gets infected,  
 17 he/she can decide to upload ID’s he/she was using the last 14 days to the server. Phones  
 18 of all users periodically download the list of ID’s which belong to the infected users and  
 19 does the matching locally.

### 20 A.3. Bluetooth-GPS apps

21 Apps that deploy both Bluetooth and GPS technology are rare. One app of this kind is  
 22 the *Aarogya Setu app* [4], developed by National Informatics Centre that comes under the  
 23 Ministry of Electronics and Information Technology, Government of India. Aarogya Setu  
 24 is following the centralized approach, and is one of the world’s fastest growing applica-  
 25 tions. The app mainly uses proximity data and GPS data are recorded only once in 30  
 26 minutes. The location data is mainly used to identify the locations where you might have  
 27 caught the infection and identify potential hotspots that may be developing when multiple  
 28 infected people visit the same place. Interaction between users is recorded by exchange  
 29 of Device Identification Numbers (DiD’s) which are static. Contact tracing data is kept on  
 30 the phone. Council of Medical Research (ICMR) shares the list of Covid-19 positive per-  
 31 sons with the Aarogya Setu server, and information about contact tracing is uploaded to  
 32 the server only if you are tested positive. The central server is then responsible for alerting  
 33 your close contacts.

### 34 References

- 35 1. Abbas, R., Michael, K.: COVID-19 Contact Trace App Deployments: Learnings From Aus-  
 36 tralia and Singapore. *IEEE Consumer Electronics Magazine* 9(5), 65–70 (2020)
- 37 2. Ahmed, N., Michelin, R.A., Xue, W., Ruj, S., Malaney, R., Kanhere, S.S., Seneviratne, A.,  
 38 Hu, W., Janicke, H., Jha, S.K.: A Survey of COVID-19 Contact Tracing Apps. *IEEE Access* 8,  
 39 134577–134601 (2020)
- 40 3. Altshuler, T.S., Hershkovitz, R.A.: Digital Contact Tracing and the Coronavirus: Israeli and  
 41 Comparative Perspectives (2020)
- 42 4. Basu, S.: Effective Contact Tracing for COVID-19 Using Mobile Phones: An Ethical Analy-  
 43 sis of the Mandatory Use of the Aarogya Setu Application in India. *Cambridge Quarterly of*  
 44 *Healthcare Ethics* 30(2), 262–271 (2021)

- 1 5. Börger, E., Stärk, R.F.: Abstract State Machines. A Method for High-Level System Design  
2 and Analysis. Springer (2003), [http://www.springer.com/computer/swe/book/  
3 978-3-540-00702-9](http://www.springer.com/computer/swe/book/978-3-540-00702-9)
- 4 6. Camus, A.: La peste. Gallimard (1947)
- 5 7. Castelluccia, C., Bielova, N., Boutet, A., Cunche, M., Lauradoux, C., Métayer, D.L., Roca, V.:  
6 ROBERT (ROBust and privacy-presERving proximity Tracing protocol). Tech. rep., Inria and  
7 Fraunhofer AISEC (2020), <https://hal.inria.fr/hal-02611265>
- 8 8. Cheng, X., Yang, H., Krishnan, A.S., Schaumont, P., Yang, Y.: KHOVID: interoperable privacy  
9 preserving digital contact tracing. CoRR abs/2012.09375 (2020), [https://arxiv.org/  
10 abs/2012.09375](https://arxiv.org/abs/2012.09375)
- 11 9. ETSI: Comparison of existing pandemic contact tracing systems. Tech. Rep. DGS E4P-002  
12 (2021), work in progress
- 13 10. ETSI: Pandemic proximity tracing systems: Interoperability framework. Tech. Rep. DGS E4P-  
14 007 (2021), v1.0.1 draft
- 15 11. Gurevich, Y.: Evolving algebras 1993: Lipari guide. In: Börger, E. (ed.) Specification and vali-  
16 dation methods, pp. 9–36. Oxford University Press (1993)
- 17 12. Hoepman, J.H.: A Critique of the Google Apple Exposure Notification (GAEN) Framework.  
18 ArXiv abs/2012.05097 (2020)
- 19 13. Huang, J., Yegneswaran, V., Porras, P., Gu, G.: On the privacy and integrity risks of contact-  
20 tracing applications (2020)
- 21 14. Liquori, L., Tedeschi, C., Vanni, L., Bongiovanni, F., Ciancaglini, V., Marinkovic, B.: Synapse:  
22 A scalable protocol for interconnecting heterogeneous overlay networks. In: Crovella, M.,  
23 Feeney, L.M., Rubenstein, D., Raghavan, S.V. (eds.) NETWORKING 2010, 9th International  
24 IFIP TC 6 Networking Conference, Chennai, India, May 11-15, 2010. Proceedings. Lecture  
25 Notes in Computer Science, vol. 6091, pp. 67–82. Springer (2010), [https://doi.org/  
26 10.1007/978-3-642-12963-6\\_6](https://doi.org/10.1007/978-3-642-12963-6_6)
- 27 15. Marinković, B., Ciancaglini, V., Ognjanović, Z., Glavan, P., Liquori, L., Maksimović, P.: Ana-  
28 lyzing the exhaustiveness of the synapse protocol. Peer Peer Netw. Appl. 8(5), 793–806 (2015),  
29 <https://doi.org/10.1007/s12083-014-0293-z>
- 30 16. Marinković, B., Glavan, P., Ognjanović, Z.: Proving properties of the chord protocol using the  
31 ASM formalism. Theor. Comput. Sci. 756, 64–93 (2019), [https://doi.org/10.1016/  
32 j.tcs.2018.10.025](https://doi.org/10.1016/j.tcs.2018.10.025)
- 33 17. Marinković, B., Liquori, L., Ciancaglini, V., Ognjanović, Z.: A distributed catalog for digitized  
34 cultural heritage. In: Gusev, M., Mitrevski, P. (eds.) ICT Innovations 2010 - Second Interna-  
35 tional Conference, ICT Innovations 2010, Ohrid, Macedonia, September 12-15, 2010. Revised  
36 Selected Papers. Communications in Computer and Information Science, vol. 83, pp. 176–186  
37 (2010), [https://doi.org/10.1007/978-3-642-19325-5\\_18](https://doi.org/10.1007/978-3-642-19325-5_18)
- 38 18. Marinković, B., Ognjanović, Z., Glavan, P., Kos, A., Umek, A.: Correctness of the chord  
39 protocol. Comput. Sci. Inf. Syst. 17(1), 141–160 (2020), [https://doi.org/10.2298/  
40 CSIS181115017M](https://doi.org/10.2298/CSIS181115017M)
- 41 19. Martin, T., Karopoulos, G., Hernández-Ramos, J.L., Kambourakis, G., Fovino, I.N.: Demysti-  
42 fying COVID-19 Digital Contact Tracing: A Survey on Frameworks and Mobile Apps. Wire-  
43 less Communications and Mobile Computing 2020(8851429), 29 (2020), [https://www.  
44 hindawi.com/journals/wcmc/2020/8851429/](https://www.hindawi.com/journals/wcmc/2020/8851429/)
- 45 20. Maymounkov, P., Mazières, D.: Kademia: A peer-to-peer information system based on the  
46 XOR metric. In: Druschel, P., Kaashoek, M.F., Rowstron, A.I.T. (eds.) Peer-to-Peer Systems,  
47 First International Workshop, IPTPS 2002, Cambridge, MA, USA, March 7-8, 2002, Revised  
48 Papers. Lecture Notes in Computer Science, vol. 2429, pp. 53–65. Springer (2002), [https://  
49 doi.org/10.1007/3-540-45748-8\\_5](https://doi.org/10.1007/3-540-45748-8_5)
- 50 21. Montagni, I., Roussel, N., Thiébaud, R., Tzourio, C.: The French Covid-19 contact tracing  
51 app: knowledge, attitudes, beliefs and practices of students in the health domain. medRxiv

- 1 (2020), <https://www.medrxiv.org/content/early/2020/11/12/2020.10.23.20218214>
- 2
- 3 22. eHealth Network: Interoperability specifications for cross-border transmission chains between  
4 approved apps (2020)
- 5 23. Ocheja, P., Cao, Y., Ding, S., Yoshikawa, M.: Quantifying the privacy-utility trade-offs in  
6 COVID-19 contact tracing apps (2020)
- 7 24. Stoica, I., Morris, R.T., Karger, D.R., Kaashoek, M.F., Balakrishnan, H.: Chord: A scalable  
8 peer-to-peer lookup service for internet applications. In: Cruz, R.L., Varghese, G. (eds.) Pro-  
9 ceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architec-  
10 tures, and Protocols for Computer Communication, August 27-31, 2001, San Diego, CA, USA.  
11 pp. 149–160. ACM (2001), <https://doi.org/10.1145/383059.383071>
- 12 25. Stoica, I., Morris, R.T., Liben-Nowell, D., Karger, D.R., Kaashoek, M.F., Dabek, F., Balakrish-  
13 nan, H.: Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM*  
14 *Trans. Netw.* 11(1), 17–32 (2003), <https://doi.org/10.1109/TNET.2002.808407>
- 15 26. Tang, Q.: Privacy-preserving contact tracing: current solutions and open questions (2020)
- 16 27. Taylor, L., Sharma, G., Martin, A., Jameson, S. (eds.): Data justice and COVID-19: Global  
17 perspectives. Meatspace Press (aug 2020)
- 18 28. Troncoso, C., et al.: Decentralized Privacy-Preserving Proximity Tracing. Tech. rep., École  
19 Polytechnique Fédérale de Lausanne, ETH Zurich, KU Leuven, Delft University of Tech-  
20 nology, University College London, Helmholtz Centre for Information Security, University  
21 of Torino, ISI Foundation (2020), [https://github.com/DP-3T/documents/blob/  
22 master/DP3T%20White%20Paper.pdf](https://github.com/DP-3T/documents/blob/master/DP3T%20White%20Paper.pdf)
- 23 29. Vukolic, M.: On the interoperability of decentralized exposure notification systems. *CoRR*  
24 *abs/2006.13087* (2020), <https://arxiv.org/abs/2006.13087>

25 **Silvia Ghilezan** is a Professor of mathematics with the University of Novi Sad and  
26 Mathematical Institute of the Serbian Academy of Sciences and Arts. On several occa-  
27 sions she has held visiting positions at University of Oregon, École Normale Supérieure  
28 de Lyon, Université Paris Diderot - Paris 7, University of Turin, Radboud University and  
29 McGill University. The major lines of her research are in mathematical logic with appli-  
30 cation to programming languages, concurrency and mathematical linguistics. Her current  
31 research interests include formal methods for new challenges in privacy protection and ar-  
32 tificial intelligence. She has collaborated with over seventy co-authors on publications in  
33 leading scientific journals and conferences (POPL, LPAR, TLCA, PDP), books and edi-  
34 torials. She acts as a SC member of FSCD, an advisor for ARVR enterprises and industry,  
35 a popularizer of science and a promoter of gender balance in science. She was awarded  
36 the distinction Chevalier (2013) and Officier (2021) de l’Ordre des Palmes Académiques  
37 of the French Republic.

38 **Simona Kašterović** is a teaching assistant at the Faculty of Technical Sciences, Uni-  
39 versity of Novi Sad. She received her B.Sc. degree at the Faculty of Sciences, University  
40 of Novi Sad in 2015. In 2017 she received her M.Sc. degree at the Faculty of Technical  
41 Sciences, University of Novi Sad. Currently, she is a Ph.D. student in applied mathemat-  
42 ics at the same faculty. In 2018 she spent three months as visiting researcher at University  
43 Paris Diderot (Paris 7) in Paris, France. Her research interests include mathematical logic  
44 and its application in computer science: proof theory, lambda calculus, type theory; un-  
45 certain reasoning; probabilistic logic; computer assisted mathematical reasoning, formal  
46 methods for artificial intelligence.

47 **Luigi Liquori** got his MS in 1990 at Udine University, Italy. He got his Ph.D. in 1996  
48 at University of Turin, Italy, and his H.d.R. in 2007 at Institut National Polytechnique



1 de Lorraine, France. He served as Lecturer at the Ecole Nationale des Mines de Nancy  
2 from 1999. Since 2001, he is a senior researcher at French Institute for Research in Com-  
3 puter Science and Automation. His research's fields range from logics and foundations  
4 of interactive proof assistants, to semantics of object oriented programming languages,  
5 until foundations of overlay networks, IoT protocols, and recently digital contact tracing  
6 against Covid-19.

7 **Bojan Marinković** got his PhD during 2014 at The Faculty of Technical Sciences  
8 University of Novi Sad, Serbia. Currently, he is Data Architect at Clarivate, Serbia. Un-  
9 til October 2018, he has been Research Assistant Professor at Mathematical Institute of  
10 the Serbian Academy of Sciences and Arts, however still interested in research in the  
11 following domains: distributed systems, applications of non-classical mathematical logic  
12 in computer science, and digitization of cultural and scientific heritage. During 2009, he  
13 spent three months as visiting researcher at INRIA Sophia Antipolis, France. **Zoran Ogn-**  
14 **janović** is a research professor at the Mathematical Institute of the Serbian Academy of  
15 Sciences and Arts. He received his PhD degree in mathematical logic from University  
16 of Kragujevac, Serbia, in 1999. He has authored or coauthored two monographs, and a  
17 number of technical papers in major international journals and conferences. His research  
18 interests concern: applications of mathematical logic in computer science, artificial intel-  
19 ligence and uncertain reasoning, automated theorem proving, applications of heuristics to  
20 satisfiability problem, and digitization of cultural and scientific heritage. He is a recipient  
21 of the Serbian Academy of Sciences and Arts Award in the field of mathematics and re-  
22 lated sciences for 2013 and the annual award of Serbian Ministry of Science for results in  
23 fundamental research in 2004.

24 **Tamara Stefanović** is a teaching assistant at the Faculty of Technical Sciences, Uni-  
25 versity of Novi Sad, Serbia. She received her B.Sc. degree at the Faculty of Sciences,  
26 University of Novi Sad in 2016. In 2019 she received her M.Sc. degree at the same fac-  
27 ulty. Currently, she is a Ph.D. student in applied mathematics at the Faculty of Technical  
28 Sciences, University of Novi Sad. Her current scientific focus is on mathematical logic  
29 and its application in computer science, especially mathematical models for data privacy.