



HAL
open science

Federating Digital Contact Tracing using Structured Overlay Networks

Silvia Ghilezan, Simona Kašterović, Luigi Liquori, Bojan Marinković, Zoran Ognjanović, Tamara Stefanović

► **To cite this version:**

Silvia Ghilezan, Simona Kašterović, Luigi Liquori, Bojan Marinković, Zoran Ognjanović, et al.. Federating Digital Contact Tracing using Structured Overlay Networks. 2021. hal-03127890v1

HAL Id: hal-03127890







<https://inria.hal.science/hal-03127890v1>

Preprint submitted on 1 Feb 2021 (v1), last revised 13 Oct 2021 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Federating Digital Contact Tracing using Structured Overlay Networks

Silvia Ghilezan ^{1,2,*} , Simona Kašterović ² , Luigi Liquori ⁴ , Bojan Marinković ^{3,1} , Zoran Ognjanović ¹  and Tamara Stefanović ² 

¹ Mathematical Institute of the Serbian Academy of Sciences and Arts, Belgrade, Serbia
² Faculty of Technical Sciences, University of Novi Sad, Serbia
³ Clarivate, Serbia
⁴ INRIA & Université Côte d’Azur, France
 * Correspondence: gsilvia@uns.ac.rs

Abstract: In this paper we present a comprehensive, yet simple, extension to the existing systems used for Digital Contacts Tracing in Covid-19 pandemic. The extension, called BAC19, will enable those systems, regardless of their underlying protocol, to enhance their sets of traced contacts and to improve 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; DCT; SON; DHT; Bluetooth; GPS

Contents

11	1 Introduction	2
12	1.1 Problem	2
13	1.2 Contributions	2
14	1.3 Overlay networks in a nutshell.	3
15	1.4 Organization of the paper	3
16	2 Digital Contact Tracing Applications	3
17	2.1 Classification by contact-tracing technology	3
18	2.2 Classification by system architecture	3
19	2.3 Geolocation-based apps.	4
20	2.4 Bluetooth-based apps	4
21	3 Overlay Networks	5
22	4 System BAC19	6
23	5 Discussion	9
24	6 Conclusions	9
25	A DCT apps - overview	10
26	A.1 Geolocation-based DCT apps	10
27	A.2 Bluetooth-based DCT apps	10
28	A.3 Bluetooth-GPS apps	11
29	References	11

Citation: Ghilezan, S.; Kašterović, S.; Liquori, L.; Marinković, B.; Ognjanović, Z.; Stefanović, T. Federating DCT using SONs. *Mathematics* **2021**, *1*, 0.
<https://dx.doi.org/>

Received:
 Accepted:
 Published:

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Submitted to *Mathematics* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

30 1. Introduction

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

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

43 *Manual Contact Tracing.* Contact tracing has mostly been done manually since many
44 centuries ago just by taking note on a simple piece of paper the list of persons and
45 goods you get in contacts (see e.g. *La Peste* by A. Camus [1]). In the actual days, manual
46 contact tracing could be exploited using simple telephone calls. Identifying contacts
47 is done through an interview with the person infected with the virus. Each person
48 is then contacted by phone. Health department should quickly alert people who are
49 close contacts that they may have been exposed to the virus. The sooner the contacts
50 are notified, the lower the risk of the spreading further. However, due to the highly
51 contagious nature of the SARS-CoV-2 virus and the fact that symptoms can manifest after
52 many days (or even never, see *asymptomatic cases*), manual contact tracing does not give
53 satisfactory results. Health departments and authorities do not have enough employees
54 to do manual contact tracing. It must be further emphasized that the SARS-CoV-2 can be
55 transmitted not only by direct contact, but also by indirect contact. The reason is that
56 infected people can leave virus droplets on any physical object they touch. In this case,
57 manual contact tracing is ineffective. For the reasons stated above, digital contact tracing
58 has been considered already at the beginning of the Covid-19 pandemic.

59 1.1. Problem

60 There is a plethora of digital contract tracing applications in use all over the world
61 fighting the Covid-19 pandemics [2]. They are developed on very different paradigms,
62 centralized [3] vs. decentralized [4], GPS based (very few indeed because of a clear
63 violation of privacy) vs. Bluetooth Low Energy based (the majority). The rush to
64 make these applications work in the shortest time led to their great diversity. The most
65 important open problem is their interoperability. There are many ongoing efforts to make
66 a federation of these different system. Herein, we address this problem and propose a
67 solution based on mathematical models of overlay networks.

68 1.2. Contributions

69 We develop a formal federation overlay network, called *BubbleAntiCovid19* (BAC19),
70 for connecting different digital contact tracing applications, which are currently in use
71 all over the world. The model is based on the well-known model of Structured Overlay
72 Networks protocols like e.g. Chord [5,6], Kademlia [7], Synapse [8]. We prove that
73 BAC19 provides a complete and fully exhaustive retrieving procedure of people that get
74 in touch with other people having tested positive to the Covid-19 disease. Hence, BAC19
75 is proven to be a simple yet powerful *interconnection* of already existing digital contact
76 tracing applications that - by construction - do not communicate with each others as
77 such providing their efficient interoperability.

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

80 1.3. *Overlay networks in a nutshell.*

81 Structured Overlay Networks are suitable model of scalable and efficient organisa-
82 tion of resources on the Internet. They represent logical organisation, independent on
83 underlying network infrastructure that physically connects available assets. Overlay
84 networks have been proven as very resilient tool in the situation when some parts of the
85 underlying infrastructure fail or become overloaded or corrupted.

86 1.4. *Organization of the paper*

87 The rest of the paper is organized as follows. Section 2 presents classifications of
88 digital contact tracing applications. Section 3 reviews Chord and Synapse protocols of
89 overlay networks. Section 4 introduces the BAC19 system and proves the completeness
90 and full exhaustiveness of the retrieving procedure. Section 5 presents a discussion
91 on other proposals for providing interoperable frameworks for digital contact tracing.
92 Section 6 concludes the paper. Appendix A gives an overview of different digital contact
93 tracing applications that are in current use against the pandemics.

94 2. **Digital Contact Tracing Applications**

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

- 106 • Contact-tracing technology;
- 107 • System architecture.

108 More information about the classification of existing contact tracing apps can be found
109 in [9,10].

110 2.1. *Classification by contact-tracing technology*

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

- 116 • Absolute location apps – Apps that track the absolute location of their users are
117 mostly based on GPS technology. Location data is stored in the form of geolocation
118 coordinate pair. These apps are also known as Geolocation-based DCT apps.
- 119 • Relative location apps – Apps that track relative location of their users are mostly
120 based on Bluetooth technology. These apps are also known as proximity DCT apps.
121 A boarding pass or a ticket for a specific event can also be considered as relative
122 location data. In order to use this kind of data, some contact tracing apps deploy
123 QR code technology.

124 2.2. *Classification by system architecture*

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

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

133 It should be stated that some generally decentralized apps have some centralized aspects
134 mainly related to user registration.

135 We will provide an overview of the existing contact tracing applications based on
136 the above classifications in Appendix A. The results are summarized in Figure 2 which
137 is motivated by [9].

138 2.3. Geolocation-based apps.

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

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

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

154 Nevertheless, geolocation-based DCT apps have also disadvantages. One of the major
155 challenges is privacy concerns, which cause low adoption rate of these applications.
156 User's privacy can be violated in several ways. Recording all user's trajectories can
157 result in revealing user's personal information such as identity, home address, work
158 address, the identity of the patient and revealing user's exposure risk to other users.
159 These problems have been elaborated in more details in [11].

160 2.4. Bluetooth-based apps

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

168 Depending on whether the exposure risk is calculated by the central server or the
169 user's device, we have centralized and decentralized apps, respectively (see Figure 1
170 motivated by [12]).

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

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

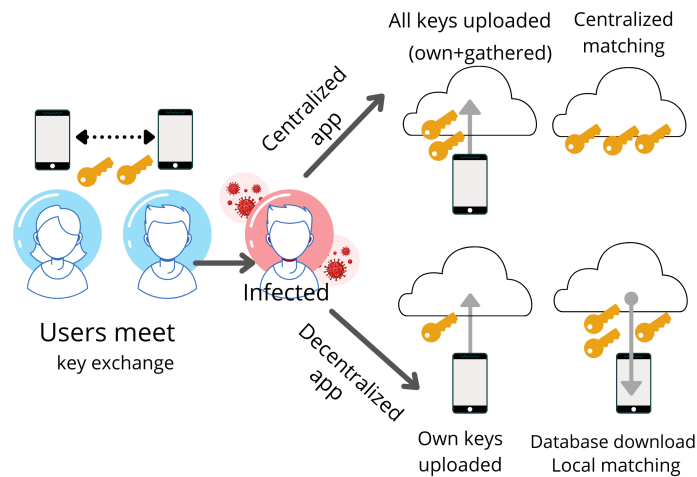


Figure 1. Centralized vs Decentralized bluetooth-based apps

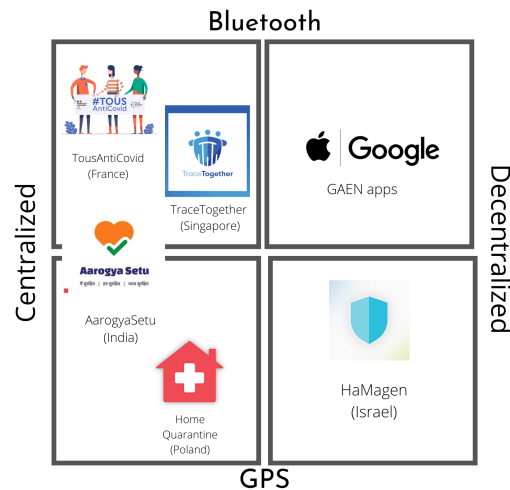


Figure 2. Classification of analyzed applications

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

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

186 3. Overlay Networks

187 As mentioned in Section 1, overlay networks are the way to organize available
188 assets.

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

194 The correctness and efficiency of the Chord's protocol lookup procedure was in
195 the focus of several papers, e.g. [5,6,13,14]. However, these properties will not be in the
196 focus of our paper. Our goal is to deliver information of every affected node, so we will
197 not use any presented improvements to speed up the process of getting results, but to
198 linearly pass every node in a Chord network, to be sure that no information is missed.

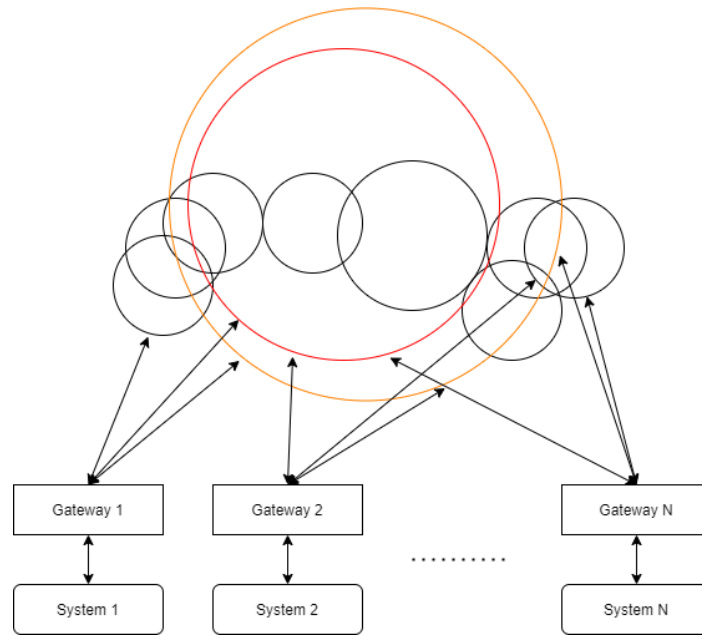


Figure 3. BAC19 Federation Overlay Network

199 Interconnection of several overlay networks is a very hard problem since different
 200 networks may use different protocols, and even in the case of several DHT networks that
 201 use the same protocol (e.g. Chord) it is enough that every overlay network uses *its own*
 202 *hash function* and information between two of them cannot be exchanged. A proposal
 203 to solve this issue was given by defining the Synapse protocol in [8]. Its performances
 204 were analyzed in [15], whereas one real-life proof of concept was developed in [16]. For
 205 the purpose of this paper we will consider the so called, *white-box* version of the Synapse
 206 protocol that, in short, allow to consider all the keys as they were using the same hash
 207 table (see [8] for details). Again, since we will use linear search procedure in one Chord
 208 network we can be sure that information will be retrieved if it exists in the system.

209 4. System BAC19

210 *BAC19* federation consists of several Chord networks:

- 211 • a network for each person/device of his/hers first contacts (black circles in Figure
 212 3),
- 213 • dedicated *red* network to connect all infected persons (red circle in Figure 3),
- 214 • dedicated *amber* network to connect all the first contacts of infected persons (amber
 215 circle in Figure 3),

216 and

- 217 • *Gateways* (black rectangles in Figure 3) as the connections to the existing digital
 218 contact tracing systems (black rounded corners rectangles in Figure 3).

219 The first connection between the proposed extension and an existing system for
 220 contact tracing is called *Gateway*. The purpose of a *Gateway* is to maintain communication
 221 between two parts and to transform messages in a way that both sides can communicate
 222 efficiently.

223 The most important thing is to maintain the mappings between identifiers (IDs)
 224 used on both sides of a *Gateway*. As we could see in Section 2, some systems periodically
 225 change IDs, so the possibility to trace those changes is vital for functioning of our system.
 226 Regarding IDs, our goal is to have one identifier per one person/device regardless of
 227 how many systems it appears in. We argue that this is possible to achieve. First, it
 228 is possible to use sufficiently large codomain of the hash function (e.g. 2^{128}). Also, it
 229 is possible to select enough parameters of a person/device so that it can be uniquely


```

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))
end

```

Algorithm 1: FindSuccessor

230 identified. We are not storing any other attribute of a person/device except newly
231 introduced identifier in our extension.

232 More precisely, with respect to the specifications that are provided in [13,15] we
233 need to introduce the following changes:

- 234 • the set $Network = \{red, amber, net_1, \dots, net_N\}, N \in \mathbb{N}$, to denote all possible net-
235 works, where N is the number of possible persons/devices in the proposed exten-
236 sion;
- 237 • the set $Time$ and the function $contact_time() : (Chord \cup \{amber\}) \times Chord \rightarrow Time$
238 to denote the time of the contact between two persons;
- 239 • the external function $current_date()$ to get the current date.

240 To obtain *completeness* and *full exhaustiveness* of the retrieval procedure the rule FIND-
241 SUCCESSOR, which finds a responsible node for a given ID, is changed as in the way
242 presented in the above Algorithm 1. With this proposal we are not compromising per-
243 formances of the extension by much. Since number of contacts of a person is relatively
244 small, it is manageable to allow increasing the complexity of the worst case retrieval from
245 $O(\log N)$ to $O(N)$. In the predefined time-slots our extension will receive the following
246 information from a system:

- 247 • all identified infected cases since the last import,

```

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
  | Set  $contacttime(amber, id) = t$ 
end

```

Algorithm 2: Put

- 248 • all confirmed cases that are not infected anymore since the last import,

```

For all inf  $\in$  Healed
  Invoke LEAVE Of Network red for inf
Algorithm 3: Leave

```

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

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

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

Algorithm 4: Set contact time

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

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
 end

Algorithm 5: Leave of network

267 During the opposite way of communication, BAC19 will pass on information to
 268 all nodes in the *amber* network to *Gateways*. If an identifier is recognized in the set of
 269 mappings for the particular origin system, the corresponding information is transferred
 270 to the origin system to alert (if not already) the person/device that she/it had risk contact
 271 with an infected person at stored timestamp. Also, BAC19 is capable to send information
 272 on the second level contacts (the result is stored in the set *Result*):

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

Algorithm 6: Get all nodes

273 Namely, for all nodes of the *amber* network it is possible to go through every origin
 274 bubble and pass those identifiers to the *Gateways*. Then the origin systems can inform
 275 those persons that they should increase their awareness since they are second level
 276 contacts.

277 Using the results from [13,15] we prove the following statement:

278 **Theorem 1.** *The proposed extension stores and retrieves only up-to-date information on Covid-*
 279 *19 positive cases (identified by the origin systems) and their contacts and makes it available to all*
 280 *origin systems.*

281 **Proof Sketch.** The execution of the proposed extension is performed in the controlled
 282 environment. Due to the scheduled time intervals for execution of different tasks, the
 283 nodes' leaving from the bubbles will not happen during the unstable states, i.e. only
 284 scenarios defined with [13, Theorems 5.3 and 5.4] will be allowed. Also, using the fact
 285 that the rule FINDSUCCESSOR is changed, all nodes will be contacted during the search
 286 procedure. Thus, the retrieving procedure of the Synapse protocol [15] is complete and
 287 fully exhaustive. □

288 5. Discussion

289 The paper [17] proposes building a common API. This approach is rather similar
290 to the extension proposed by this paper. However, these approaches have also two
291 significant differences:

- 292 • while [17] building API connection points between each of two different origin
293 systems that are connected, our extension proposes a version to common bus where
294 each of the origin systems communicate with the proposed extension and in this
295 way reduces and simplifies number of connection points that needs to be maintained
296 when several origin systems are connected;
- 297 • with BAC19 we are simplifying also information that is being exchanged, and we
298 do not violate privacy in the origin systems (since our extension does not collect
299 information of origin DCT system).

300 A guideline on Interoperability specifications for cross-border transmission chains be-
301 tween approved apps by the European Community [18] proposes a Federation Gateway
302 Service for synchronizing the diagnosis keys (keys of infected users) across backend
303 servers of each national app. However, this approach focuses only on Google/Apple
304 exposure notification apps because the majority of European countries have developed
305 this kind of apps, and also because one Google/Apple exposure notification app can
306 detect the contact with a user of another Google/Apple exposure notification app. In this
307 paper we do not focus on a certain type of DCT apps, we want to achieve the connection
308 between them regardless the contact-tracing technology and their system architecture.
309 We leave to the the reader to envisage the following scenario:

- 310 • Alice lives in the region which has centralized DCT *System A*, while Bob lives in
311 the region which has centralized DCT *System B*. Bob has spent some time in the
312 region A, and both of them are traveling together side by side with negative RT-PCR
313 tests. However, Bob developed symptoms of Covid-19 after couple of days and was
314 confirmed as positive.

315 If *System A* and *System B* are part of BAC19, it would be enough that only one of Alice
316 and Bob had installed system from the other region just in the time of travel for Alice to
317 be informed that she is the first contact of a infected person.

318 6. Conclusions

319 In this paper we have presented BAC19 a new and efficient overlay network con-
320 necting existing systems for digital contact tracing. The advantages of BAC19 (its usage)
321 are:

- 322 • a person does not install anything new on his/her mobile device (except a new
323 application which is used in the region that this person is visiting);
- 324 • the overlay does not store any personal sensitive information;
- 325 • the overlay is independent regarding how the origin system calculated contacts or
326 is it based on Bluetooth or GPS technology;
- 327 • the overlay supports manual entry of recognized contacts;
- 328 • there are no new highly complicated calculations of possible contacts beside those
329 that are performed by the original contact tracing systems.

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

333 **Author Contributions:** These six authors contribute equally to this paper. Part of this paper
334 was written during the the period of selfisolation of one of the authors due to the contact with a
335 Covid-19 positive person.

336 **Funding:** This work was partly supported by: the Science Fund Republic of Serbia #6526707
337 AI4TrustBC.

338 **Conflicts of Interest:** The authors declare no conflict of interest.

339 **Abbreviations**

340 The following abbreviations are used in this manuscript:

341	BLE	Bluetooth Low Energy
	DCT	Digital Contact Tracing
342	GPS	Global Positioning System
	DHT	Distributed Hash Table
	SON	Structured Overlay Network

343 **Appendix A. DCT apps - overview**

344 *Appendix A.1. Geolocation-based DCT apps*

345 *Home Quarantine.* At the beginning of the COVID-19 pandemic, Ministry of Digital
346 Affairs of Poland developed the Home Quarantine app [19]. This is a typical example
347 of a centralized app which deploys GPS technology. It is developed to support the
348 authorities, especially the police and social services, with adequate information about
349 people undergoing mandatory home quarantine. Users are also required to upload their
350 digital photos. So, aside the GPS technology the app also uses face recognition. The
351 app is mandatory for anyone who has developed coronavirus symptoms. It should be
352 emphasized that Poland also developed the ProteGO Safe app for alerting users of close
353 contact with an infected person based on The (Google/Apple) Exposure Notification
354 (GAEN) system.

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

364 *Appendix A.2. Bluetooth-based DCT apps*

365 *Blue-Trace protocol apps.* Singapore's Government Technology Agency in collabo-
366 ration with Ministry of Health in March 2020 released the TraceTogether app [21] that
367 allows digital contact tracing using the custom BlueTrace protocol. Australia has later
368 adopted the protocol and released the CovidSafe app [22]. Contact tracing is done using
369 Bluetooth Low Energy and proximity data is encrypted and stored only on the users
370 phone. Users in the contact log are identified using anonymous time-shifting "temporary
371 IDs". If a user tests positive for the infection, the Ministry of Health requests his/her
372 contact log. The user has the right to choose whether to share the contact log or not. If
373 the user chooses to share the log, the contact log is uploaded to a central server and the
374 health authority is then responsible for matching the log to contact detail and informing
375 close contacts of the infected user. These apps are examples of Bluetooth-based semi-
376 centralized apps. It should also be noted that Singapore solved the problem of tracing
377 people who don't use smartphones by enabling the app to work with Token - a physical
378 Bluetooth-based device.

379 *ROBERT protocol app.* The French National Assembly released the StopCovid app
380 in May 2020. The app has later been renamed TousAntiCovid [23]. It allows digital
381 contact tracing using the ROBust and privacy-presERving proximity Tracing protocol
382 (ROBERT protocol). It also deploys Bluetooth technology and belongs to the category of
383 semi-centralized apps. The difference between this app and apps based on the BlueTrace
384 protocol relates to confirmation of positive users. More precisely, in France when a

385 person is confirmed to be positive, the lab gives a patient a QR code and the scanned
386 code is the proof for the app that you are infected. It is up to you to share this information
387 with the app, and if you choose to share this information with a central server, the server
388 is responsible for alerting your close contacts.

389 *Google/Apple exposure notification apps.* In April 2020 Google and Apple announced
390 the joint work on decentralized Bluetooth-based protocol named The (Google/Apple)
391 Exposure Notification (GAEN) system [24]. Many states then developed different apps
392 using the Google/Apple Exposure Notification framework including Austria (Stopp
393 Corona app), Germany (Corona-Warn-App), Italy (Immuni), Canada (COVID Alert) etc.
394 The principle by which applications work is as follows. During a close contact, user's
395 phones exchange random Bluetooth identifiers. These identifiers change frequently and
396 the information about exchanged ID's is stored on the user's phone. When a user gets
397 infected, he/she can decide to upload ID's he/she was using the last 14 days to the
398 server. Phones of all users periodically download the list of ID's which belong to the
399 infected users and does the matching locally.

400 *Appendix A.3. Bluetooth-GPS apps*

401 Apps that deploy both Bluetooth and GPS technology are rare. One app of this kind
402 is the *Aarogya Setu app* [25], developed by National Informatics Centre that comes under
403 the Ministry of Electronics and Information Technology, Government of India. Aarogya
404 Setu is following the semi-centralized approach, and is one of the world's fastest growing
405 applications. The app mainly uses proximity data and GPS data are recorded only once
406 in 30 minutes. The location data is mainly used to identify the locations where you might
407 have caught the infection and identify potential hotspots that may be developing when
408 multiple infected people visit the same place. Interaction between users is recorded by
409 exchange of Device Identification Numbers (DiD's) which are static. Contact tracing data
410 is kept on the phone. Council of Medical Research (ICMR) shares the list of COVID-19
411 positive persons with the Aarogya Setu server, and information about contact tracing
412 is uploaded to the server only if you are tested positive. The central server is then
413 responsible for alerting your close contacts.

References

1. Camus, A. *La peste*; Gallimard, 1947.
2. E4P, E. Comparison of existing pandemic contact tracing systems. Technical report, ETSI, 2021.
3. Castelluccia, C.; Bielova, N.; Boutet, A.; Cunche, M.; Lauradoux, C.; Métayer, D.L.; Roca, V. ROBERT (ROBust and privacy-preserving proximity Tracing protocol). Technical report, Inria and Fraunhofer AISEC, 2020.
4. et al., C.T. Decentralized Privacy-Preserving Proximity Tracing. Technical report, École Polytechnique Fédérale de Lausanne, ETH Zurich, KU Leuven, Delft University of Technology, University College London, Helmholtz Centre for Information Security, University of Torino, ISI Foundation, 2020.
5. Stoica, I.; Morris, R.T.; Karger, D.R.; Kaashoek, M.F.; Balakrishnan, H. Chord: A scalable peer-to-peer lookup service for internet applications. Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August 27-31, 2001, San Diego, CA, USA; Cruz, R.L.; Varghese, G., Eds. ACM, 2001, pp. 149–160. doi:10.1145/383059.383071.
6. Stoica, I.; Morris, R.T.; Liben-Nowell, D.; Karger, D.R.; Kaashoek, M.F.; Dabek, F.; Balakrishnan, H. Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Trans. Netw.* **2003**, *11*, 17–32. doi:10.1109/TNET.2002.808407.
7. Maymounkov, P.; Mazières, D. Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. Peer-to-Peer Systems, First International Workshop, IPTPS 2002, Cambridge, MA, USA, March 7-8, 2002, Revised Papers; Druschel, P.; Kaashoek, M.F.; Rowstron, A.I.T., Eds. Springer, 2002, Vol. 2429, *Lecture Notes in Computer Science*, pp. 53–65. doi:10.1007/3-540-45748-8_5.
8. Liquori, L.; Tedeschi, C.; Vanni, L.; Bongiovanni, F.; Ciancaglini, V.; Marinkovic, B. Synapse: A Scalable Protocol for Interconnecting Heterogeneous Overlay Networks. NETWORKING 2010, 9th International IFIP TC 6 Networking Conference, Chennai, India, May 11-15, 2010. Proceedings; Crovella, M.; Feeney, L.M.; Rubenstein, D.; Raghavan, S.V., Eds. Springer, 2010, Vol. 6091, *Lecture Notes in Computer Science*, pp. 67–82. doi:10.1007/978-3-642-12963-6_6.
9. Ocheja, P.; Cao, Y.; Ding, S.; Yoshikawa, M. Quantifying the Privacy-Utility Trade-offs in COVID-19 Contact Tracing Apps, 2020, [arXiv:cs.CY/2012.13061].
10. Tang, Q. Privacy-Preserving Contact Tracing: current solutions and open questions, 2020, [arXiv:cs.CR/2004.06818].

11. Cheng, X.; Yang, H.; Krishnan, A.S.; Schaumont, P.; Yang, Y. KHOVID: Interoperable Privacy Preserving Digital Contact Tracing. *CoRR* **2020**, *abs/2012.09375*, [2012.09375].
12. Huang, J.; Yegneswaran, V.; Porras, P.; Gu, G. On the Privacy and Integrity Risks of Contact-Tracing Applications, 2020, [arXiv:cs.CR/2012.03283].
13. Marinkovic, B.; Glavan, P.; Ognjanovic, Z. Proving properties of the Chord protocol using the ASM formalism. *Theor. Comput. Sci.* **2019**, *756*, 64–93. doi:10.1016/j.tcs.2018.10.025.
14. Marinkovic, B.; Ognjanovic, Z.; Glavan, P.; Kos, A.; Umek, A. Correctness of the Chord protocol. *Comput. Sci. Inf. Syst.* **2020**, *17*, 141–160. doi:10.2298/CSIS181115017M.
15. Marinkovic, B.; Ciancaglini, V.; Ognjanovic, Z.; Glavan, P.; Liquori, L.; Maksimovic, P. Analyzing the exhaustiveness of the Synapse protocol. *Peer Peer Netw. Appl.* **2015**, *8*, 793–806. doi:10.1007/s12083-014-0293-z.
16. Marinković, B.; Liquori, L.; Ciancaglini, V.; Ognjanović, Z. A Distributed Catalog for Digitized Cultural Heritage. ICT Innovations 2010 - Second International Conference, ICT Innovations 2010, Ohrid, Macedonia, September 12-15, 2010. Revised Selected Papers; Gusev, M.; Mitrevski, P., Eds., 2010, Vol. 83, *Communications in Computer and Information Science*, pp. 176–186. doi:10.1007/978-3-642-19325-5_18.
17. Vukolic, M. On the Interoperability of Decentralized Exposure Notification Systems. *CoRR* **2020**, *abs/2006.13087*, [2006.13087].
18. eHealth Network. Interoperability specifications for cross-border transmission chains between approved apps, 2020.
19. Ministry of Foreign Affairs Republic of Poland, Home Quarantine. <https://www.gov.pl/web/diplomacy/home-quarantine-monitoring-by-taketask> Accessed January 10, 2021.
20. Israel ministry of health, HaMagen. <https://govextra.gov.il/ministry-of-health/hamagen-app/download-en/> Accessed January 9, 2021.
21. Government of Singapore, TraceTogether. <https://www.tracetogether.gov.sg/> Accessed January 9, 2021.
22. Government of Australia, CovidSafe. <https://www.covidsafe.gov.au/> Accessed January 9, 2021.
23. Government of France, TousAntiCovid. <https://gitlab.inria.fr/stopcovid19> Accessed January 10, 2021.
24. Hoepman, J.H. A Critique of the Google Apple Exposure Notification (GAEN) Framework. *ArXiv* **2020**, *abs/2012.05097*.
25. Government of India, Aarogya Setu. <https://aarogyasetu.gov.in/> Accessed January 10, 2021.