Efficient Anonymous Group Communication

Vom Fachbereich Informatik
der Technischen Universität Darmstadt genehmigte

Dissertation

zur Erlangung des akademischen Grades
Doctor rerum naturalium (Dr. rer. nat.)

Eingereicht von:
Tim Grube
geboren in Seeheim-Jugenheim

Tag der Einreichung:	15. Mai 2018
Tag der Disputation:	10. Juli 2018
Erstreferent:	Prof. Dr. Max Mühlhäuser
Korreferent:	Prof. Dr. Mathias Fischer

Fachgebiet Telekooperation
Fachbereich Informatik
Technische Universität Darmstadt
Hochschulkennziffer D 17

Darmstadt 2018

Tim Grube: *Efficient Anonymous Group Communication,*
Darmstadt, Technische Universität Darmstadt
Jahr der Veröffentlichung der Dissertation auf TUprints: 2018

Things are only impossible until they're not.

— Captain Jean-Luc Picard,
Star Trek: The Next Generation, "When The Bough Breaks" (1988)

Dedicated to the loving memory of my father Volker Grube
1958 – 2007

ABSTRACT

This dissertation addresses the important challenge of efficiency in anonymous communication. Solving this challenge is essential to provide anonymity in group communication.

Every exchanged message leaks metadata: this information describes the communication itself with, among others, sender, recipients, frequency of the communication. While the law protects this information, it is often published and misused with consequences for the participants of the communication—often consequences particular for the senders of information.

Anonymous communication systems like *Tor* break the link between senders and recipients of messages and diminish emerging metadata. However, their design requires duplicating messages for all recipients early, mostly at the sender itself. With that, the system has to handle an unnecessary burden of processing identical messages. This dissertation contributes a novel mechanism that establishes communication groups such that the message duplication is pushed as close to the recipients as possible. This dissertation also shows that this efficiency improvement does not come at costs of anonymity. Moreover, the group establishment mechanism increases the robustness of the communication against users that leave and join the communication system. To encounter the additional information leakage, different mechanisms to share routing information are introduced and discussed under the angle of efficiency and anonymity. To also protect senders of messages, this dissertation adapts Dining-Cryptographer networks to enable sender anonymity with an adjustable trade-off between efficiency and anonymity.

CONTRIBUTIONS The scientific contributions of this dissertation fall into the following categories:

1. *Efficient Communication Overlays*: A novel overlay establishment mechanism is presented. This mechanism adapts *Ant Colony Optimization* (ACO) to connect senders and recipients with a reduced number of connections while the anonymity remains stable.

2. *Reliable Communication under Churn*: Churn often disrupts communication overlays, this thesis proposes a mechanism to counter this disruptions and increase the robustness of the communication. For that, the ACO-based mechanism utilizes residual pheromones to reconnect subjects to the communication overlay.

3. *Routing Information Exchange considering Efficiency and Anonymity*: Four methods to share routing information, namely successor lists, successor lists with multi-layer encryption, Bloom filters, and distributed lookup tables, are introduced into the anonymous communication setting, discussed and evaluated with respect to their properties concerning efficiency and anonymity.

4. *Efficient and Effective Sender Protection*: A novel mechanism based on *asymmetric dining-cryptographer networks (ADCnets)* is proposed to improve the efficiency of sender protection without degrading anonymity over time. Moreover, the trade-off between efficiency and anonymity can be controlled.

EVALUATION The developed mechanisms have been extensively evaluated using a combination of simulations and formal arguments. For this evaluation, a graph-based simulation model has been developed enabling to analyze the improvement of the ACO-based communication overlays over conventional overlays. An extensive simulation identified valuable parameter combinations for the ACO mechanism, leading to communication overlays with an efficiency improvement of up to 40%. This efficiency improvement increases the communication delay only by up to 2 extra hops. The calculation of the achieved anonymity degree indicates no loss of anonymity in comparison to conventional overlays; even more, the anonymity degree does even improve. Under churn, the robustness of the communication also increases by 30% in comparison to conventional overlays.

The four approaches to routing information exchange are discussed and compared using formal arguments. The evaluation enables to select the appropriate mechanism based on the requirements for memory and communication efficiency and anonymity for the desired application scenario.

The novel adapted ADCnets enable sender anonymity with configurable efficiency and anonymity. As such, they are supe-

rior to current approaches implementing cover traffic. A simulation study shows that cover traffic randomization improves efficiency at the cost of anonymity. The proposed ADCnets have been evaluated using formal arguments that demonstrate the efficiency improvement and the preservation of anonymity. As both efficiency and anonymity are conflicting, they cannot be achieved at the same time; however, the proposed ADCnets enable to balance efficiency and anonymity to the requirements of the application scenario.

ZUSAMMENFASSUNG

Diese Dissertation adressiert die Steigerung der Effizienz anonymer Kommunikation um diese massentauglich zu machen.

Bei jeglicher Kommunikation entstehen Metadaten, welche den Prozess des Nachrichtenaustausches beschreiben, beispielsweise mit Informationen zu Sender, Empfängern, sowie der Intensität der Kommunikation. Metadaten sind durch Gesetze geschützt, trotzdem werden sie immer wieder veröffentlicht und unbefugt genutzt mit oft unerwünschten Konsequenzen – insbesondere für den Sender der Informationen.

Anonyme Kommunikationssysteme wie zum Beispiel Tor reduzieren Metadaten: durch die Weiter- und Umleitung der Nachrichten wird die „sichtbare" Verbindung zwischen Sender und den Empfängern unterbrochen. Aufbau und Funktionsweise anonymer Kommunikationssysteme erfordert zumeist die frühzeitige Duplikation versendeter Nachrichten für jeden einzelnen Empfänger; in der Regel erfolgt die Duplikation durch den Sender selbst. Hierdurch entsteht eine unnötige Menge identischer Nachrichten im Kommunikationssystem. Es wird ein neuer Mechanismus zur Erstellung effizienterer Kommunikationsstrukturen eingeführt; dieser verschiebt die Duplikation der Nachrichten so nah wie möglich an die Empfänger. Es wird gezeigt, dass die daraus resultierende Effizienzsteigerung nicht zu Lasten der Anonymität geht. Neben der Effizienzsteigerung wird auch die Robustheit gegenüber Nutzern gesteigert, die das System dynamisch verlassen und es wieder betreten. Hierdurch müssen Routing-Informationen häufiger ausgetauscht werden; um dem zu begegnen werden verschiedene Verfahren aus den Blickwinkeln Effizienz und Anonymität betrachtet. Zuletzt führt diese Dissertation ein neues Verfahren zum Schutz der Senderanonymität ein; hierfür werden Dining-Cryptographer Netzwerke (DC-Netze) angepasst und deren Effizienz gesteigert. Das Verfahren ermöglicht die dynamische Abwägung von Effizienz und Anonymität.

BEITRÄGE Die wissenschaftlichen Beiträge dieser Dissertation sind in die folgenden Kategorien eingeordnet:

1. *Effiziente Kommunikationsstrukturen*: Basierend auf Ameisenkolonie-Optimierung (*ant colony optimization*, ACO) werden effizientere Kommunikationsstrukturen erstellt, welche die Anzahl involvierter Verbindungen minimieren; hierdurch wird die Anonymität nicht beeinträchtigt.

2. *Robuste Kommunikation*: Dynamische Teilnehmer (*churn*) unterbrechen Kommunikationsstrukturen regelmäßig. Um diesen Unterbrechungen entgegenzutreten wird das etablierte Wissen (d.h. die Pheromone der Ameisen) genutzt um Teilnehmer schneller wieder mit der Kommunikationsgruppe zu verbinden.

3. *Austausch von Routing-Informationen*: Es werden vier Methoden zum Austausch von Routing-Informationen vorgestellt: Pfadlisten mit und ohne mehrschichtige Verschlüsselung, Bloom-Filter und verteilte Suchtabellen. Diese werden diskutiert und bezüglich ihrer Effizienz- und Anonymitätseigenschaften beurteilt.

4. *Effizienter und effektiver Schutz von Sendern*: Basierend auf dem Konzept *asymmetrischer DC-Netze* (ADC-Netze) wird ein Verfahren zum effizienteren Schutz der Anonymität der Sender eingeführt. Dieses Verfahren verhindert die Beeinträchtigung der Anonymität im Verlauf der Zeit. Zudem ermöglicht es eine dynamische Abwägung von Effizienz und Anonymität.

AUSWERTUNG Die entwickelten Mechanismen wurden mit Hilfe einer Kombination von Simulationen und formaler Diskussion ausgewertet. Für diese Evaluation wurde ein Graphbasiertes Simulationsmodell entwickelt und implementiert. Dabei wurden die ACO-basierten Kommunikationsstrukturen mit konventionellen Kommunikationsstrukturen verglichen. Eine ausführliche Simulation bewertete relevante Konfigurationsparameter des ACO-Verfahrens; dies führte zu Kommunikationsstrukturen die bis zu 40% effizienter sind. Zudem konnte gezeigt werden, dass die Verzögerung, die durch die Effizienzsteigerung erzeugt wird, der Kommunikation mit bis zu zwei weiteren Weiterleitungen begrenzt ist. Es konnte weiterhin gezeigt werden, das kein Anonymitätsverlust durch die Effizienzsteigerung bedingt wird, sondern die erreichbare Anonymität noch gesteigert

werden kann. Unter dem Einfluss dynamischen Nutzerverhaltens konnte die Robustheit um 30% im Vergleich zu konventionellen Kommunikationsstrukturen auf Basis kürzester Pfade gesteigert werden. Vier Ansätze zum Austausch von Routing-Informationen wurden eingeführt und in einer formalen Diskussion unter Berücksichtigung von Speicher- und Kommunikationseffizienz und Anonymität verglichen. Der neue auf ADC-Netzen basierende Ansatz zum Schutz der Anonymität des Senders von Nachrichten ermöglicht die dynamische Abwägung von Effizienz und Anonymität. Hierdurch ist dieser Ansatz dem heutigen Schutz durch zusätzlich eingefügte sinnfreie Nachrichten (*cover traffic*) überlegen. In einer weiteren Simulation wurde gezeigt, dass eine Effizienzsteigerung dieses Schutzverfahrens mit einem allmählichen Anonymitätsverlust einhergeht. Die vorgeschlagenen ADC-Netze wurden mittels formaler Diskussion evaluiert; hierbei konnte gezeigt werden, dass diese die Effizienz steigern können während die Anonymität über die Zeit konstant gehalten werden kann. Der Zielkonflikt zwischen Effizienz und Anonymität verhindert das gleichzeitige Erfüllen beider Ziele; ADC-Netze ermöglichen jedoch das dynamische abwägen zwischen diesen Zielen.

PUBLICATIONS

1. Böck, L., Karuppayah, S., Grube, T., Mühlhäuser, M. & Fischer, M. *Hide and seek: Detecting sensors in P2P botnets* in *2015 IEEE Conference on Communications and NetworkSecurity (CNS)* (IEEE, Sept. 2015), 731–732. ISBN: 9781467378765. doi:10.1109/CNS.2015.7346908.
2. Daubert, J., Fischer, M., Grube, T., Schiffner, S., Kikiras, P. & Mühlhäuser, M. AnonPubSub: Anonymous publish-subscribe overlays. *Computer Communications* **76,** 42–53. ISSN: 01403664 (Feb. 2016).
3. Daubert, J., Grube, T., Mühlhäuser, M. & Fischer, M. *Internal attacks in anonymous publish-subscribe P2P overlays* in *2015 International Conference and Workshops on Networked Systems (NetSys)* (IEEE, Mar. 2015), 1–8. ISBN: 9781479958047. doi:10.1109/NetSys.2015.7089074.
4. Daubert, J., Grube, T., Mühlhäuser, M. & Fischer, M. *On the anonymity of privacy-preserving many-to-many communication in the presence of node churn and attacks* in *2016 13th IEEE Annual Consumer Communications and Networking Conference (CCNC)* (IEEE, Jan. 2016), 738–744. ISBN: 9781467392921. doi:10.1109/CCNC.2016.7444871.
5. Grube, T., Hauke, S., Daubert, J. & Mühlhäuser, M. *Ant colonies for efficient and anonymous group communication systems* in *2017 International Conference on Networked Systems (NetSys)* (IEEE, Mar. 2017), 1–8. ISBN: 9781509043941. doi:10.1109/NetSys.2017.7903958. <http://ieeexplore.ieee.org/document/7903958/>.
6. Grube, T., Hauke, S., Daubert, J. & Mühlhäuser, M. *Ant colony optimisation - A solution to efficient anonymous group communication?* in *2017 14th IEEE Annual Consumer Communications and Networking Conference (CCNC)* (IEEE, Jan. 2017), 337–340. ISBN: 9781509061969. doi:10.1109/CCNC.2017.7983129.

7. Grube, T., Schiller, B. & Strufe, T. *Monotone sampling of networks* in *2nd International Workshop on Dynamic Networks and Knowledge Discovery (DyNaK)* **1229** (CEUR Workshop Proceedings, Sept. 2014), 37–48.

8. Grube, T., Thummerer, M., Daubert, J. & Mühlhäuser, M. *Cover Traffic: A Trade of Anonymity and Efficiency* in *Security and Trust Management* (eds Livraga, G. & Mitchell, C.) (Springer International Publishing, 2017), 213–223. ISBN: 978-3-319-68063-7.

9. Grube, T., Volk, F., Mühlhäuser, M., Bhairav, S., Sachidananda, V. & Elovici, Y. *Complexity Reduction in Graphs: A user Centric Approach to Graph Exploration* in *International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC)* (IARA, 2017), 24–31.

10. Grube, T., Volk, F., Mühlhäuser, M., Bhairav, S., Sachidananda, V. & Elovici, Y. User-guided Graph Exploration: A Framework for Algorithmic Complexity Reduction in Large Data Sets. *International Journal On Advances in Intelligent Systems* **11,** 68–80. ISSN: 1942-2679 (June 2018).

11. Karuppayah, S., Böck, L., Grube, T., Manickam, S., Mühlhäuser, M. & Fischer, M. *SensorBuster: On Identifying Sensor Nodes in P2P Botnets* in *International Conference on Availability, Reliability and Security (ARES)* (ACM, Sept. 2017), 34:1–34:6.

12. Valipour, S., Volk, F., Grube, T., Böck, L., Karg, L. & Mühlhäuser, M. *A formal holon model for operating future energy grids during blackouts* in *5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)* (IEEE, Apr. 2016), 146–153. ISBN: 9789897581847.

There's no point in being nuts if you can't have a little fun.

— John Nash
A Beautiful Mind (2001)

ACKNOWLEDGMENTS

This thesis would not have come to realization without the support and encouragement of my advisors, colleagues, friends, and family. My gratitude goes out to all of them.

First, my warmest thanks to my advisor Max Mühlhäuser for his continuous support, encouragement, confidence and advice which allowed me to reach this final stage of writing this thesis. Second, I am also grateful to my second referee Mathias Fischer for the supervision and all of the discussions that challenged and inspired me.

A big thank you goes to my colleagues from the Telecooperation Lab and the RTG 2050, especially the colleagues and former colleagues from the Security and Protection in Networks group (and its predecessors SSI and SST): Jörg Daubert, Rolf Egert, Carlos Garcia, Emmanouil Vasilomanolakis, Sheikh M. Habib, Sascha Hauke, Florian Volk, Andrea Tundis, Leon Böck, Nikolaos Alexopoulos, and Aidmar Wainakh. You made the office a second home; keep up the spirit! The same goes to the staff of the Telecooperation Lab: Elke Halla, Nadine Moldaner, Silke Romero-Ostermann, Wiebke Kronz, Elke Reimund, Fabian Herrlich, and Sebastian Alles.

I also won't forget the time and start of my journey in the Peer-to-Peer group around Thorsten Strufe, Benjamin Schiller, Hani Salah, Stefanie Roos, Thomas Paul, and Giang Nguyen. Also a big thank you for the input and the discussions to Peter Buxmann as my Tandem-PI in the RTG 2050 and to the visiting guest researchers Stephen Marsh and Esma Aïmeur for their valuable input.

Last but not least, I would like to thank my family and friends. Without you, I would not have been able to complete this thesis. Especially thank you to my wife Johanna for your encouragement and unconditional love.

CONTENTS

Appendix

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ACO	ant colony optimisation
ADCnet	asymmetric dining-cryptographer network
apl	average shortest path length
DCnet	dining-cryptographer network
DoS	denial of service
DHT	distributed hash table
IPC	Inter-process communication
OSN	Online Social Network
OR	onion routing

P2P	peer-to-peer
Pub/Sub	Publish/Subscribe
RGG	random geometric graph
RPC	remote procedure call
TTL	time to live
StA	Steiner tree approximation
WSN	wireless sensor network

1 INTRODUCTION

> [The Human Rights Council recognizes ...] *that the right to privacy can enable the enjoyment of other rights and the free development of an individual's personality and identity, and an individual's ability to participate in political, economic, social and cultural life, and noting with concern that violations or abuses of the right to privacy might affect the enjoyment of other human rights, including the right to freedom of expression and to hold opinions without interference, and the right to freedom of peaceful assembly and association [...]*
>
> — UN, Human Rights Council
> `A/HRC/34/L.7 Rev.1`

1.1 MOTIVATION & PROBLEM STATEMENT

With the advancing connectivity of devices of all kinds, for example, smartwatches, smartphones, and smart home devices, digital communication is becoming ubiquitous, and communication often becomes *internet-based* services. Not only the increasing number of connected devices but in particular *always online* devices like smartphones foster ubiquitous communication.

(Digital) Communication concentrates on just a handful of services. For example, Facebook had 2.2 billion monthly active users as of the fourth quarter of 2017. With that, Facebook achieves a worldwide market penetration of 26.3% as of June 2017; the market penetration for North America is 72.4%, for Europe 41.7%. With WhatsApp and the Facebook Messenger, Facebook owns two of the three most popular messaging services (WhatsApp ranks first, Facebook Messenger third)[1].

Ride-sharing is another exemplary internet-based service. Here, a similar concentration becomes visible with Uber (having an estimated market share of 77% in the USA) and Lyft (having an estimated market share of 20%) [76].

With this concentration of communication to a small selection of services, the service providers can collect (and consequently monetize) immense amounts of data [101, p. 19]. Thus, service providers can connect data sets across different services and gain additional insights [101, p. 15f]. These data are then often used to

[1] Source: `www.statista.com`

perform targeted advertising, thereby financing the supposedly *free* services for the users.

Collection and usage of data are regulated, for example, by the EU's general data protection regulation [55] and its predecessor [54]. According to the UNESCO [101] and the EU [54], processing of data should only be performed in such a way is it would be expected by the information supplier (the user). The inability—or unwillingness—of service providers to follow this general rule is, for example, shown by Uber: in 2012, Uber released a blog post where they analyzed their database to identify so-called "rides of glory", rides towards potential one-night stands [144].

Users mostly are unaware of this data analysis, continuous hacks and leaks of personal and sensitive information [88, 120], and probably unintended consequences of (anonymized) data publication as, for example, Strava's heatmap[2] [14, 89, 95]. The continuity of these undesired events shows that service providers are not willing—or able—to protect the user's data.

While these privacy issues seem to be tolerable for some users, others are impaired by information leaks and advanced data analysis; this is particularly pressing when considering that often no illegal activity is conducted yet the harm for people may be considerable.

To overcome emerging privacy threats, several attempts to anonymize communication [15, 30, 37, 46, 61, 134, 146] have been made. In a first step, these systems try to remove the service provider as a central point with full data access by decentralia-tion [15, 30, 146], i.e., assigning the functionality of communication establishment to multiple entities operated by different subjects, or distribution of this functionality to all participants [37, 46, 61]. However, these approaches fail to provide sufficient protection of users due to efficiency-issues caused by different communication schemes and anonymization mechanism. Currently used and proposed anonymization systems [15, 46] mostly focus on point-to-point communication whereas communication often occurs in a many-to-many scheme. These mismatching communication schemes lead to excessive signaling overhead or require centralized mediation services that arrange the communication more efficiently at the cost of anonymity and privacy. Also, the anonymization mechanisms often rely on cover traffic and ob-

[2] https://labs.strava.com/heatmap

fuscation, both requiring plenty of additional messages to be effective.

Systems that tackle privacy-preserving and anonymous communication for groups (many-to-many) either focus on access control [10] and confidentiality [134] or do not utilize the full potential of the used Publish/Subscribe (Pub/Sub) scheme [37].

Particularly in this application scenario of ubiquitous many-to-many communication, privacy and anonymity are not the only desirable properties. Increasing heterogeneity also increases the need for efficiency in the establishment of the communication. From smartwatches to servers, weak devices, such as smartwatches, that participate in many-to-many communication are less capable than, for example, computers or even servers.

1.2 THIS THESIS

The goal of this thesis is to provide novel methods of *efficient* and anonymous group communication. The following research questions will be answered in this thesis:

- *Which group establishment mechanism is able to consider efficiency and anonymity? To what extent is the trade-off between these two optimization goals controllable?*

- *Can the communication system react to dynamic users, i.e., react to users that join and leave the communication system, while minimizing message loss and maintaining overall efficiency and anonymity?*

- *To what extent can the information leakage caused by routing information exchange be limited? What is the resulting impact on the overall communication efficiency?*

- *Which technique can protect senders of messages from de-anonymization while considering both efficiency and anonymity?*

EFFICIENT AND ANONYMOUS GROUP COMMUNICATION Early approaches to privacy-preserving many-to-many communication [122, 134, 141] focus on providing confidentiality. I.e., they focus on hiding information from adversaries that are not in possession of key material that would enable them to participate in the communication system. More recent work by Daubert et al. [37–40] focusses on providing anonymity *and* confidentiality; here, also internal adversaries are considered, and protection mechanisms that alter the communication structures are presented. These systems have in common that they rely on rather inefficient, flooding-based communication overlays [37–40] or cannot protect the anonymity of the users [122, 134, 141].

The first contribution of this thesis is a novel approach for establishing efficient yet anonymous communication groups. For that, Chapter 4 presents a mechanism based on ant colony optimisation (ACO) that enables more efficient communication groups by utilizing message aggregation and late message duplication. The adoption of ACO to the establishment of anonymous communication overlays is described in a multi-phase approach that establishes and optimizes communication groups. The influence of the parameters of the ACO-based overlay establishment mechanism is evaluated in detail. The evaluation compares the resulting ACO-based communication overlays to state of the art flooding-based communication overlays.

Figure 1 shows an example of the difference between traditional flooding-based shortest path communication overlays on the left-hand side and ACO-based communication overlays on the right-hand side; subject 3 is a sender, subjects 5, 8, and 9 are recipients. The ACO-based communication overlay aggregates the path towards recipients; with this, the system reduces the number of connections and moves the point of message duplication towards recipients. Thereby, the system lessens the number of identical messages that are necessary to reach all recipients.

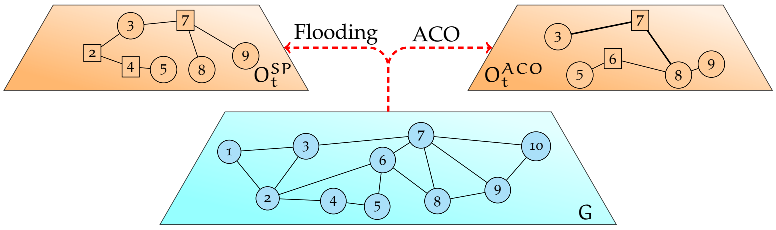


Figure 1: Higher Efficiency Communication Groups by applying ACO-based Overlay Establishment

This novel ACO-based overlay establishment mechanism improves the anonymous communication model of Daubert et al. [37] and has been published with a detailed analysis in [67, 68].

ROBUST COMMUNICATION UNDER CHURN Distributed, peer-to-peer (P2P)-based communication systems suffer from churn, i.e., suffer from leaving and joining subjects. Leaving subjects may disrupt communication overlays and require potentially expensive repair to reconnect all participants to the communication overlay. Figure 2 visualizes such an example: subject 7 is responsible for connecting the sender 3 to the recipients 5, 8, and 9; when subject 7 leaves the system, it disrupts the communication overlay, i.e., the recipients cannot receive messages from the sender. The system has to repair the overlay, here, by establishing the alternative path using subjects 2 and 4.

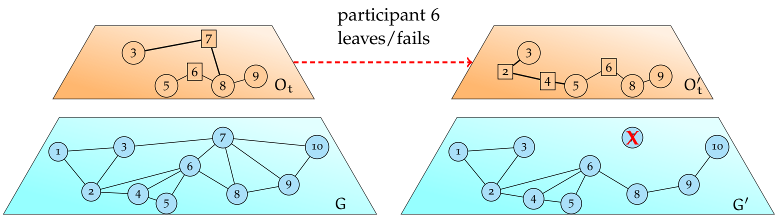


Figure 2: Coping with Churn using ACO-based mechanisms

While many related approaches focus on static snapshots for their analysis [37, 146], i.e., churn is often ignored, the influence of churn is undisputed and still a challenge [137, 151, 152].

Chapter 5 presents in its first part how the novel ACO-based overlay establishment mechanism enables the underlying communication system to recover from leaving users and improves the performance for joining users. This improvement is achieved by utilizing residual path markers and using them to connect newly joining participants and repair the communication overlay when participants leave the system. The utilization of residual path markers enables fast repairs by biasing the agents that establish the communication paths.

ROUTING AS A TRADE-OFF Particularly in dynamic systems with subject churn, subjects need to exchange routing information more frequently. Routing of messages requires exchanging information between the subjects in the communication system. The more information is exchanged, the better the efficiency of the communication overlay, for example, by establishing optimal dissemination paths. This information exchange, however, also degrades anonymity as structural information about the communication overlays as well as information about the roles of the participants is shared amongst all other subjects.

As a result of this, efficiency concerning communication and memory, and anonymity of the routing information exchange gains importance. Communication overhead is here related to the amount of structural information that has to be included in every message; naturally, a second option is to have the knowledge distributed to all subjects to avoid sending it around. Anonymity relates to the evidence about senders and recipients that is leaked by the routing information. Chapter 5 compares and evaluates the effect that four different mechanisms of sharing routing information have on both efficiency and anonymity in its second part.

EFFICIENT AND EFFECTIVE SENDER PROTECTION Protecting the anonymity of senders of information is a hard problem. Obfuscation through noise causes large communication overhead as it has to be in place continuously. Altering the topology to conceal the sender does not solve the problem, from a global perspective like the service provider's perspective, the sender will still be the source of a message.

Current systems that include sender anonymity utilize various mechanisms to obfuscate the senders of messages [15, 30, 34, 58, 146]. All these mechanisms have in common that they rely on cover traffic, i.e., they blur communication messages and, therefore, conceal the sender in noise. However, such cover traffic increases the communication overhead which renders the systems unusable for participants with fewer resources.

This thesis proposes a sender protection mechanism that is based on dining-cryptographer networks (DCnets) [22]. There are several improvements introduced to the classical DCnet approach to improve the communication overhead:

1. each DCnet spans only a relatively small subgroup (of configurable size) and not all participants of the communication system,
2. the small(er) DCnet group is used to hide the sender of a message only (instead of using it for the whole communication), and
3. the communication within such a group is directed towards a single rendezvous subject who then releases

the message into the communication system (instead of broadcast-based communication).

The novel approach based on asymmetric dining-cryptographer networks (ADCnets) for sender protection provides provable anonymity based on the security guarantees of DCnets with a configurable trade-off between efficiency (smaller groups) and anonymity (larger groups). An adversary confronted with this technique cannot reduce the sender's anonymity below the group size. Figure 3 visualizes the concept of ADCnets, where the sender 4 is hidden in an ADCnet formed around the rendezvous participant RS. With this ADCnet, an adversary not able to distinguish the subjects c1–c3, 4, and RS to identify the sender of a message.

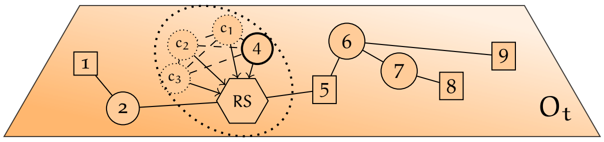


Figure 3: Sender protection using ADCnet: Sender 4 is hidden in an ADCnet around rendezvous subject RS.

Chapter 6 discusses the challenge of sender protection in details and provides insights into state of the art as well as the novel concept of ADCnets. The insights into state of the art have been published in [70].

This thesis addresses several challenges. First, the challenge of combining the opposing goals of efficiency and anonymity in the establishment of communication groups is addressed by proposing a novel mechanism establishing these communication groups. Second, the influence of dynamic user behavior ("churn") is analyzed and addressed by utilizing residual information from the initial communication group establishment. Third, the information leakage of routing information exchange mechanisms is analyzed, and four different mechanisms are compared with respect to their effects on efficiency and anonymity. Lastly, a novel mechanism for sender protection is proposed that hides senders in a group of configurable size.

The core contributions of this thesis have been mainly published in three peer-reviewed publications [67, 68, 70]. Additionally, the author of this thesis (co-)authored nine further peer-reviewed publications [18, 37, 39, 40, 69, 71, 72, 80, 145]. The list of publications can be found on Page xi.

1.3 OUTLINE

The remainder of this thesis is structured as follows: Chapter 2 establishes the background knowledge on core optimization goals of this thesis efficiency and anonymity. The chapter then continues with an introduction of the general communication model, as well as a clarification of frequently used terms. The following adversary model is derived using core properties of the adversary and provides real-world examples. Finally, Chapter 2 introduces metrics to evaluate anonymity and discusses anonymization primitives.

The state of the art in anonymous communication systems is reviewed in Chapter 3. The first part of this chapter derives a series of requirements and characteristics that define essential properties of anonymous communication systems concerning efficiency and anonymity. A detailed discussion of the start of the art follows the requirements and characteristics.

Chapter 4 presents the novel ACO-based mechanism to establish anonymous communication overlays for group communication. First, it introduces ACO itself; then it introduces the ACO-based overlay establishment mechanism and discusses an extensive simulation study that evaluates the influence of the parameters and the performance in comparison with conventional overlays and a Steiner tree approximation. Finally, a formal discussion calculates the possible anonymity protection.

Chapter 5 introduces subject churn, i.e., it discusses the challenge of joining and leaving users. Churn may disrupt the communication overlay. Another simulation study shows that the novel ACO-based overlay establishment mechanism improves the robustness against these dynamic users. The chapter concludes with a detailed discussion of mechanisms to disseminate routing information in the tension of efficiency and anonymity.

Chapter 6 discusses why sender protection is a particular challenge. Next, the chapter discusses cover traffic as the state of the art approach and analyzes efficiency and anonymity using a randomized initialization. Finally, Chapter 6 introduces ADCnets as a mechanism to protect senders in anonymous group communication with higher efficiency and abiding sender anonymity.

Chapter 7 summarizes this thesis and highlights the contributions and insights gained from the research performed. Chapter 7 concludes with an outlook on future research directions.

2 BACKGROUND

This chapter introduces concepts, terminology, and background knowledge for efficient and anonymous group communication. These concepts will then be used to discuss the state-of-the-art and the contributions in Chapters 3–6. The chapter is structured as follows: Section 2.1 introduces the notions efficiency and anonymity. Section 2.2 introduces the communication model and terminology that is used throughout this thesis. Section 2.3 contains a thorough discussion of adversary models. Section 2.4 introduces anonymity metrics as well as commonly used anonymization primitives. This chapter concludes with a summary in Section 2.5.

2.1 EFFICENCY & ANONYMITY

Efficiency and anonymity in communication systems are entangled measures with inherent tension. Efficiency measures costs of communication concerning different metrics, e.g., time or load. Anonymity measures the ability to conceal information about the communication relationship and can achieve different levels of protection. As such, efficiency improvements usually require more information and anonymity improvements require reduction of information.

This section first takes a closer look at efficiency and (native) group communication, introducing the characteristics of both. Second, the often synonymously used terms of anonymity, confidentiality, and privacy are discussed and differentiated. After that, this section introduces four levels of anonymity.

2.1.1 *Efficient Group Communication*

EFFICIENCY Efficiency in communication systems can be measured concerning time, i.e., the delay between senders and recipients, or load, i.e., the number of messages that is necessary to transmit messages between senders and recipients.

Time-efficiency is often expressed as a communication system being low-delay or high-delay [48]. The classification of anonymous communication systems into low- and high-delay is not clearly defined by the state of the art. A "typical" separation classifies anonymous communication systems that enable e-mail-like, asynchronous communication as high-delay; anonymous communication systems that enable real-time, synchronous communication like chatting or voice-communication are classified as low-delay.

The lower the delay, the higher the time-efficiency. Equation (1) formulates the time-efficiency eff_{time} as proportional to the difference between the time the message is sent t_{sent} and the time the message is received by the last recipient $t_{receive}$. This difference mostly depends on the maximal distance $distance_{max}(s, r)$ between sender s and recipient r; this thesis uses this distance as measure for the time-efficiency, i.e., expresses the time-efficiency with the diameter of a communication overlay $diam(O_t)$ (see Section 2.2.1.3 Paragraph Graph-theoretical Properties).

$$eff_{time} \propto t_{receive} - t_{send} \propto distance_{max}(s, r) \tag{1}$$

Load-efficiency is measured by the number of messages that are necessary to relay a message from its sender to its recipient(s). The lower the number of messages, the higher the load-efficiency.

With native group communication (see following paragraph (native) group communication), messages are duplicated as close to the recipients as possible, i.e., communication paths are consolidated. A Steiner tree establishes the communication with the *least number of connections*—with native group communication, Steiner trees provide the best load-efficiency by minimizing the number of required messages.

When calculating the load-efficiency, maintenance messages like heartbeats can be ignored as they depend on the communication system itself and not on ocuring communication. The load-efficiency therefore only considers communication messages and messages that are introduced by anonymity protection measures like cover traffic. Acknowledgements may be used for all communication messages, as such, they scale with both the number of sent communication messages and the minimal number of sent messages; therefore, the load-efficiency can be calculated without considering acknowledgements. Equation 2 formulates the load-efficiency eff_{load} as ratio of the number of messages that are required to establish a communication $msg_{com+prot}$

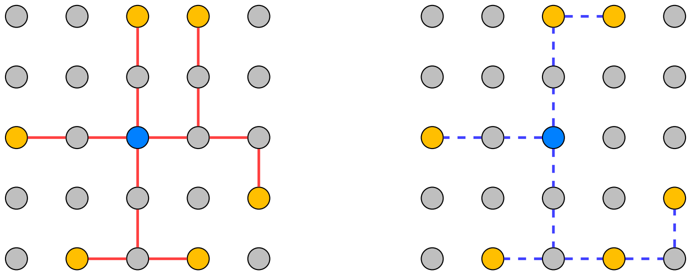


Figure 4: Efficiency may be optimized with respect to *delay* on the left-hand side or with respect to *overhead* on the right-hand side.

by the minimal number of messages that are required to establish a communication with a Steiner tree $\mathtt{msg}_{\mathtt{St+prot}}$. Both are composed of the number of messages for the communication $\mathtt{msg}_{\mathtt{com}}$ or $\mathtt{msg}_{\mathtt{St}}$ respectively and the required number of protection messages $\mathtt{msg}_{\mathtt{prot}}$. The number of protection measures $\mathtt{msg}_{\mathtt{prot}}$ may be different in both situations and depends on the employed anonymity protection measures.

$$\mathtt{eff}_{\mathtt{load}} \propto \frac{\mathtt{msg}_{\mathtt{com+prot}}}{\mathtt{msg}_{\mathtt{St+prot}}} = \frac{\mathtt{msg}_{\mathtt{com}} + \mathtt{msg}_{\mathtt{prot}}}{\mathtt{msg}_{\mathtt{St}} + \mathtt{msg}_{\mathtt{prot}}} \tag{2}$$

Example 1 shows with Figure 4 the difference between delay-optimized and load-optimized communication.

> **Example 1:** In Figure 4, the sender (blue) sends a message to the recipients (orange). On the left-hand side, the message dissemination is optimized for time-efficiency, so it uses the shortest and, thus, fastest, path. The dissemination structure on the right-hand side is optimized for load-efficiency, so it uses the minimal number of messages to reach out to all recipients.
> Assume that a message requires one time-step to be relayed one hop, the sender emits the message at time $t = 1$, no anonymity protection measures causing additional messages, and both sides employ native group communication, time-efficiency and load-efficiency values are as follows:
>
> $$\mathtt{eff}^{\mathtt{left}}_{\mathtt{time}} = 4 - 1 = 3 \qquad \mathtt{eff}^{\mathtt{left}}_{\mathtt{load}} = {}^{13}/_{11} = 1.18$$
> $$\mathtt{eff}^{\mathtt{right}}_{\mathtt{time}} = 6 - 1 = 5 \qquad \mathtt{eff}^{\mathtt{right}}_{\mathtt{load}} = {}^{11}/_{11} = 1$$

This thesis focusses on improving load-efficiency; thus, this thesis will refer to *load-efficiency* as *efficiency*. Nonetheless, time-

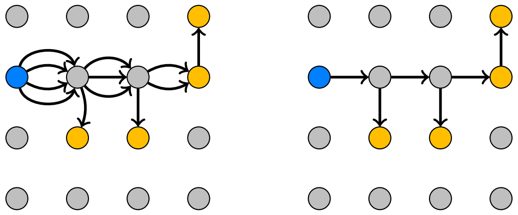


Figure 5: Native group communication significantly reduces the messaging load.

efficiency is still considered and evaluated in the context of this thesis.

(NATIVE) GROUP COMMUNICATION Group communication describes the communication scenario in which a message is sent towards multiple recipients (*1-to-n*). When multiple subjects of a group are sending messages, the scenario results in many-to-many (*m-to-n*) group communication. Often, group communication is realized by repeated unicasts (*1-to-1*), i.e., realized by having the sender emit the message for each recipient individually.

Native group communication is based on application layer multicast and improves efficiency by moving the point of message duplication as close as possible to the respective recipients (*late message duplication*). With this late message duplication, the number of messages that are required to disseminate a message to all recipients is reduced from the overall system's perspective.

> **Example 2:** By applying native group communication, as visualized in Figure 5 on the right-hand side, the load for subjects can be reduced. In the example visualized, the load is reduced by 50 %, from 12 to 6 messages.

2.1.2 *Anonymity, Confidentiality, and Privacy*

Anonymity, confidentiality, and privacy are often confused as synonyms; even the state of the art research does not provide a consistent distinction [34, 43, 118, 129, 134, 141]. This section provides definitions of the three security-related properties based on either broadly accepted knowledge (to define confidentiality) or a distinction based on the state of the art and elaboration (to define anonymity and privacy).

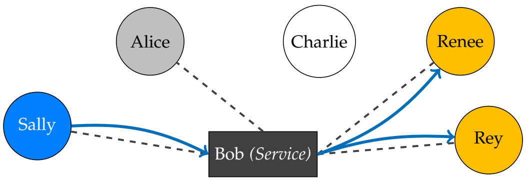


Figure 6: Example communication system. The dashed connections represent the participation in the communication system; the solid arrows represent the actual communication.

The following definitions use the following scenario which is also visualized in Figure 6: Bob offers the communication service. Sally is a sender of messages in Bob's service, and Renee and Rey are recipients of messages in Bob's service. Alice is a participant in Bob's service but is not involved in the communication of Sally, Renee, and Rey, and Charlie is not participating in Bob's service.

CONFIDENTIALITY A communication system provides confidentiality, if only intended parties, i.e., senders and recipients as specified by the participation in the communication group, can learn information about the content of a communication message. In other words, the knowledge of all other parties, including any adversary, does not change when they can access a message during its dissemination, e.g., while relaying the message: the adversary's posterior knowledge about the content equals the a priori knowledge about the content.

> **Example 3:** In Bob's confidentiality-providing communication service, only Sally, Renee, and Rey can access the contents of a message. Neither Alice nor Bob can learn anything about the content of the message. Alice, because she does not take part in the system, and Bob because he is only relaying the message without sharing the necessary cryptographic key material.

PRIVACY Hughes and Shmatikov [77] define privacy as the unobservability of *identity-based system behavior*. Thus, nobody who is probing a subject should be able to discriminate whether the subject participates in the system or not, based on the behavior of the system.

> **Example 4:** While an adversary can observe the behavior of Bob's privacy-preserving service, he should not be able to learn whether Charlie, Sally, Renee, or Rey are using the service at all by evaluating his observations, for instance, seen messages.

ANONYMITY Anonymity expresses the hiding of information regarding linking of senders, recipients, and messages. *Sender anonymity* considers the linking of senders to messages to be sensitive and measures whether an adversary can collect evidence to match a sender to a message. *Recipient anonymity* is symmetrically for recipients; thus, considers the linkage of a recipient to messages to be sensitive. *Sender-Recipient anonymity* considers the linkage of senders to recipients using a message.

Hughes and Shmatikov [77] provide a formal model, based on Kripke structures, defining anonymity in four stages (here given for the sender's perspective, recipient's and sender-recipient's perspective follow symmetrically):

- *unlinkability:* A communication system provides unlinkability if an adversary is able to identify either the sender of a message or the recipient of a message but not both for the same message. Thus, an adversary is not able to link the sender of a message with its recipient(s).

  > **Example 5:** An adversary can identify Sally as the sender or Renee and Rey as the recipients of a message; however, the adversary is not able to link Sally, and Renee or Rey simultaneously to the same message.

- *k-anonymity*[1]*:* A communication system provides k-anonymity if an adversary is able to reduce the respective candidate set of subjects to k but not further. Thus, a sender hides in at least k-1 other subjects, and each recipient hides in at least k-1 other subjects as well.

[1] Hughes and Shmatikov base their notion of k-anonymity on work of Samarati and Sweeney [128]

> **Example 6:** An adversary can reduce the set of subjects being potential senders of a communication message to size k, and Sally as the sender is included in this set. Symmetrically, the adversary can compute the set of potential recipients where Renee and Rey are then included. The adversary is not able to remove additional subjects from the candidate sets such that the minimal size is stable with k subjects.

- *absolute anonymity*: In contrast to k-anonymity, a communication system provides absolute anonymity if an adversary is not able to reduce the set of potential senders or recipients. Thus, senders and recipients hide within the set of all subjects in the communication system.

  > **Example 7:** An adversary is not able to reduce the number of potential senders beyond the number of subjects in the communication system.

- *untraceability:* A communication system provides untraceability if an adversary is unable to observe any evidence of communication taking place. Thus, the adversary is neither able to discriminate senders or recipients nor to realize that there is a communication ongoing.

  > **Example 8:** An adversary is not able to realize that Sally and Rey are communicating. This inability of detecting communication does not change, independent from the adversary being the service provider Bob, an outsider Charlie, or a different participant Alice, that is not involved in the communication.

By the order of the strength of anonymity, the following ranking of an anonymous communication is achieved, from weak to strong:

`unlinkability` $<$ `k-anonymity` $<$ `absolute anonymity` $<$ `untraceability`

2.1.3 *Levels of Anonymity*

According to the scenario of this thesis, communication systems are classified according to their level of anonymity. The level of anonymity describes the ability of the system to hide the communication relationship—sender and recipient(s)—from the adversaries of different strength, i.e., from adversaries impersonating participants, outsiders, or the service provider.

In communications, two types of information may be leaked: first, the actual content of the communication and second, the metadata of the communication.

- *Content*: The payload of messages carries the content of the communication. It may or may not include personally identifiable information, i.e., information that may be used to identify a human user [55].

- *Metadata*: During the communication *process*, metadata emerge. These metadata encompass, among others, information about senders, recipients, and frequency of communication. As such, they can be used to identify human users, and are, therefore, also personally identifiable information [55]. Simply speaking, metadata comprise all information that can be derived without utilizing the content of a message.

  Metadata can be further partitioned into *explicit* metadata and *implicit* metadata. Explicit metadata are information that help the communication system to provision its service:

  1. *Routing information* that is included *in communication messages*
  2. Information included *in maintenance messages*, e.g., route establishment messages
  3. Information included *in the infrastructure*, e.g., routing tables

  Implicit metadata are *derivable from observing the behavior of subjects*. As such, subjects may be identified as senders of messages by observing that they are the source of the respective messages.

Figure 7 visualizes the following levels of anonymity using the analogy of a letter: the content is the actual letter; metadata are included on the envelope. Explicit metadata are represented by the actual addresses, and the implicit metadata are represented by the list of forwarders.

- level-0 communication systems do not offer any protection of their users' communication and operate using *plain text* messages.

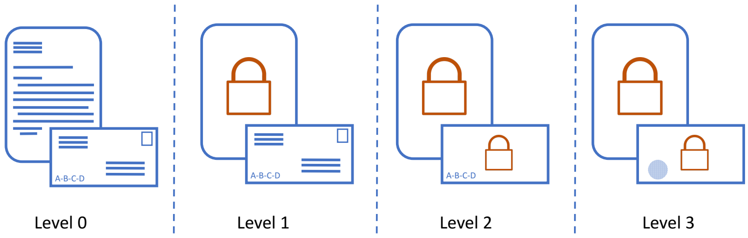


Figure 7: Levels of Anonymity Protection

- level-1 systems provide confidentiality but no anonymity. The content of the communication is hidden from unintended parties. However, the system does not provide any protection of metadata. By inspecting message headers or performing traffic analysis [123], sender and recipient(s) of a message can be identified.

> **Example 9:** Bob's communication service provides confidentiality using end-to-end encryption. However, as Bob is relaying messages between Sally and Rey, he can access the header of the message and read source and recipient fields.

- level-2 communication systems provide confidentiality and protect explicit metadata. Thus, anonymity is preserved when a non-global adversary accesses the message.

> **Example 10:** Bob's communication service provides level-2 anonymity by encrypting message headers. Thus, Alice is not able to identify Sally and Rey as sender and recipient of a message. However, as Bob has a global view of his service, he can trace the message from origination to destination and, thus, identify Sally and Rey as sender and recipient.

- level-3 communication systems provide confidentiality and protect all metadata. Thus, anonymity is not only preserved against local adversaries but also against global adversaries by hiding the message dissemination paths.

> **Example 11:** Bob's communication service provides level-3 anonymity by encrypting message headers and obfuscating message origination and destination(s) by utilizing a DC network [22].

Figure 7 visualizes the differences of the levels of anonymity. This thesis focusses on anonymous communication systems that aim to provide anonymity on level-3, as metadata threaten the anonymity of users.

2.2 MODEL AND TERMINOLOGY

Anonymous group communication requires a formalized model that encompasses the process of communication as well as its establishment. Furthermore, the link towards distributed communication based on P2P [113, 140] is provided in this section.

2.2.1 *Group Communication based on Publish/Subscribe*

This section describes how group communication is established using the Pub/Sub paradigm and provides a model that introduces the terminology and formal background for this thesis.

2.2.1.1 *Concept*

The Pub/Sub concept enables group communication based on topics or interests. Based on these topics, senders are enabled to send messages to recipients—without having senders and recipients to know each other. Recipients enable the intermediate broker (or *Event-service* according to the Pub/Sub terminology) to relay messages by expressing their interests to topics using the concept of subscriptions.

2.2.1.2 *Publish/Subscribe and Generation of Communication Groups*

Pub/Sub is a content-oriented routing scheme. In contrast to other common methods that enable distributed systems and communication, e.g., Inter-process communication (IPC), remote procedure call (RPC), and distributed objects, Pub/Sub provides loose coupling between senders and recipients. Moreover, Pub/Sub naturally supports many-to-many communication and message dissemination based on topics and interests [53].

Loose coupling enables communication between senders and recipients while those are not necessarily known to each other. A broker in between acts as a proxy between senders and recipients and relays messages to establish the communication. By reducing the coupling to a minimum, senders and recipients may remain anonymous to each other. The remaining coupling originates from the topic-based addressing scheme, were senders and recipients share a topic which is used to relay messages between them. This indirection enables both senders and recipients to blend in larger groups of subjects.

The broker enables group communication—or many-to-many communication—based on topics. For each topic, a group is established using either advertisement-based process or a subscription-based one. In the advertisement-based process, the senders disseminate advertisements for topics to which they to emit messages. Upon receiving an advertisement for a topic they are interested in, recipients reply with a subscription. By following these subscriptions, the broker can relay the messages accordingly. In the subscription-based process, senders skip advertisements and recipients disseminate their subscriptions in the communication system to enable the brokers to relay messages from the senders.

> **Example 12:** Sender Sally sends an advertisement m^t_{adv} to the broker Bob. Bob then relays the advertisement to all other subjects. Upon receiving m^t_{adv}, recipient Rey sends the subscription message m^t_{sub} to Bob. When receiving a communication message m^t_{com} from Sally, Bob forwards it to Rey by looking up preceding subscriptions.

2.2.1.3 *Communication Model*

The communication model comprises a formal representation of the system, different abstraction layers, Pub/Sub as enabling communication technique, and a differentiation of both content and metadata.

GRAPHS, PARTICIPANTS, AND ROLES A communication system can be modeled as graph G with the subjects, for instance, Sally, Rey, Alice, and Bob, represented as vertices $\mathcal{V}$ and their connections represented as edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Both human users and (smart) devices can participate in the communication system and represent subjects $v \in \mathcal{V}$. Users are typical human participants of the system. As connected (smart) devices can also participate in such communication systems, this thesis uses the general term *subjects* to abstract from unnecessary specifics of participants.

Communication groups are formed along topics $t_i \in T$. These topics represent interests of communication groups and are the characteristic attributes of these groups.

The set of *senders* $\mathcal{S}_t$ comprises information producers, for example, Sally. Accordingly, the set of *recipients* $\mathcal{R}_t$ comprises information consumers, for example, Renee and Rey. The set of *brokers* $\mathcal{B}_t$ consists of the subjects that are required to relay messages, for example, Bob. In centralized or decentralized communication systems, dedicated subjects, for example, servers, populate $\mathcal{B}_t$. In contrast, distributed communication systems require regular subjects to populate $\mathcal{B}_t$ and to relay communication traffic.

The sets $\mathcal{S}_t$, $\mathcal{R}_t$, and $\mathcal{B}_t$ are not required to be disjoint, for example, Rey can be a recipient $r_i \in \mathcal{R}_t$ and still be required to relay communication to subsequent subjects, i.e., be a broker $b_j \in \mathcal{B}_t$ with $r_i = b_j = v_k \in V$.

UNDERLAY AND OVERLAYS The subjects and their *physical* connections[2] form the *underlay* U. Subjects usually connect to dedicated servers from the service provider, when the system is a centralized or a decentralized system. These servers are then establishing the end-to-end connection between subjects by relaying messages. In distributed systems, this functionality is taken over by the subjects themselves. They are connected based on certain properties, e.g., their distances or physical connections, and cooperate to establish the service, i.e., relay messages for each other to create the end-to-end connection between senders and recipients.

The *logical underlay* G is formed above the underlay. It represents connections between subjects in the communication system and, thus, depends on the connection structure of the communication system, for example, friendships or shared interests. In P2P systems, the logical underlay G represents the physical connections, i.e., in P2P systems, the underlay U and the logical underlay G are equal.

> **Example 13:** Scrutinizing the connections on Facebook, the underlay and the logical underlay can easily be derived. The underlay is formed by the Facebook server to which subjects are connected. The logical underlay is then derived by extracting the so-called *social graph* from Facebook which contains the logical connections between the users of Facebook.

[2] This definition of physical connections ignores the internet service provider as entity as they are forwarding messages transparently.

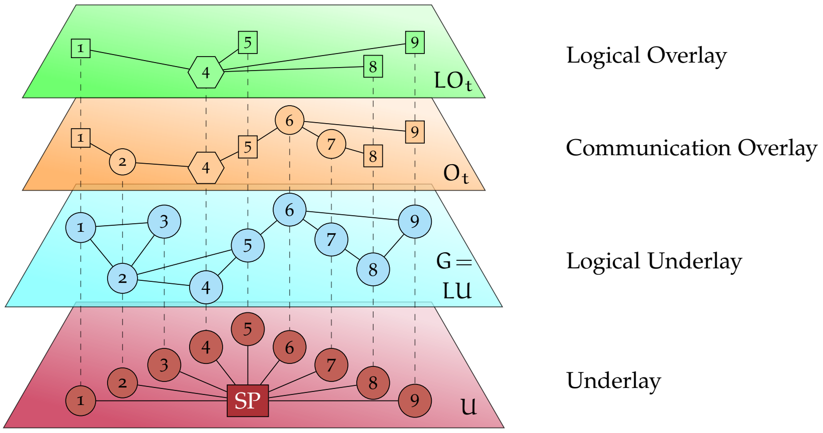


Figure 8: The layers of the communication system

For each communication, no matter whether it is a pairwise communication or a group-based one, a communication *overlay* O_t is established. This overlay comprises all participants that are communicating, i.e., the senders $s \in \mathcal{S}_t$ and recipients $r \in \mathcal{R}_t$, and the participants that are necessary to deliver information from sender to receivers, i.e., the brokers $b \in \mathcal{B}_t$. The overlay is denoted by O_t and induced by subjects $\mathcal{V}_t = \mathcal{S}_t \cup \mathcal{R}_t \cup \mathcal{B}_t$ and connections $\mathcal{E}_t \subseteq \mathcal{E}$. A communication overlay is a set of $|\mathcal{S}_t|$-*dissemination trees* with the roots being the senders $s \in \mathcal{S}_t$.

Above the communication overlay, the logical overlay LO_t is formed. This logical overlay includes the senders $\mathcal{S}_t$ and recipients $\mathcal{R}_t$, i.e., only the subjects eligible to access the content of the communication take part in the communication group form the logical overlay.

Figure 8 shows the layers in their relation to each other. Depending on the actual system design, the underlay U and the logical underlay G may be the same layer; decentralized P2P systems are often systems having equal underlay U and logical underlay G layers. Following from here, this thesis abstracts from the underlay; its contributions are based upon the logical underlay G to create and to improve the communication overlays O_t while maintaining the same, stable logical overlays LO_t.

GRAPH-THEORETICAL PROPERTIES Each subject v inhibits properties from respective graph-structures than can be derived from the different layers. Within this thesis, properties with respect to the logical underlay G and a communication overlay O_t are relevant.

In these layers, the structure of subjects and their respective connections may be represented by the graphs G and O_t. In these, each subject v_i as a number of connections which is called their *degree* $d(v_i)$. This degree can be separated by directed connections, then a subject has an *in-degree* $d^+(v_i)$ and an *out-degree* $d^-(v_i)$. The degrees count the respective incident connections of v_i. The adjacent subjects, i.e., the subjects directly connected to v_i with one of the incident connections, are called *neighbors* $N(v_i)$. In the case of directed connections, $N(v_i)$ is differentiated into $N^+(v_i)$, i.e., the adjacent subjects that can communicate towards v_i, and $N^-(v_i)$, i.e., the adjacent subjects that v_i can communicate towards.

A *path* between two subjects v_i and v_j is a *sequence of connections* $p(v_i, v_j) = ((v_i, v_a), (v_a, v_b), ...(v_c, v_j))$ with v_i being the source of the first connection and v_j being the destination of the last connection. The *length of a path* $|p(v_i, v_j)|$ is the number of connections in the sequence. A shortest path $sp(v_i, v_j)$ is such a path with the shortest sequence of connections possible; the shortest path lengths is the respective number of connections in such a path. The *average shortest path length (apl)* is the average of the shortest path length between all connected pairs of subjects. The *diameter* is the length of the longest shortest path between any connected pair of subjects.

2.2.2 *Synchronizing Subjects: Real World and Simulation*

The communication model presented in this section assumes that subjects progress in synchronized rounds, i.e., events like receiving and sending messages can be ordered globally. This ordering of events simplifies evaluation and comparison of the contributions of this thesis and the state of the art.

In a distributed system, synchronization is not easy to achieve. This model, however, mostly does not require synchronized progress. In the following, the core modules of the simulation model are discussed concerning their requirements for synchronization in the context of this thesis:

- *Overlay Establishment (Chapters 4 and 5)*: subjects follow overlay the overlay establishment protocol in a fully distributed fashion. As such, subjects process messages based on their local knowledge and the protocol. Synchronization beyond a time to live (TTL) is not required.

- *Communication (Chapters 4 and 5)*: the sender may emit a message at any time. The subjects in the respective communication overlay relay the message according to the overlay; recipients consume the message, and may further relay the message according to the overlay. Synchronization beyond a TTL is not required.

- *Cover Traffic (Chapter 6)*: cover traffic requires subjects to emit cover messages in every round. Thus, synchronization is required to ensure that cover traffic can deploy its effect. The synchronization does, however, not require that all subjects are completely synchronized; a common understanding of "rough" time intervals is sufficient. For example, it is sufficient that subjects emit their cover message with deviations from each other as long as every subject emits a message before a subject may emit a second message.

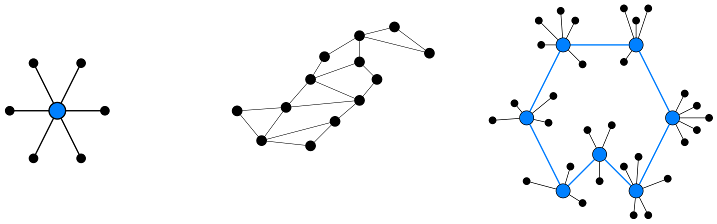
- *Dining-cryptographer networks DCnets (Chapter 6)*: DCnets rely on multi-party computation. As a result of this, all participants of a DCnet are required to contribute to every message to enable successful decryption. If a subject misses participating, the content of the message cannot be retrieved. Thus, the communication takes place in rounds which have to be agreed by the participants. However, it is not required that the subjects emit their messages at exact same points in time.

Rounds can be translated to time; sometimes, the number of elapsed rounds needs to be translated to elapsed time, for example, when analyzing the delay of communication. The number of rounds can be multiplied with the average connection delay. Projects like *PingER* [99] measure the delay between end-hosts in varying distances. Hereby, they derive average connection delays for, for example, intra-continental and inter-continental connection. These can be used to estimate the delay that is introduced by each hop.

2.2.3 *P2P Systems*

P2P systems distribute the functionality of the server(s) to all of the subjects that participate in the system. P2P systems can be grouped into the following classes of centralized and decentralized systems. The third class of hybrid systems combines both centralized and decentralized elements.

Figure 9 visualizes the three classes of P2P systems that are explained in the following.



(a) centralized P2P (b) decentralized P2P (c) hybrid P2P

Figure 9: P2P system classes. Blue vertices are central entities while black vertices are usual subjects.

CENTRALIZED P2P SYSTEMS In centralized P2P systems, a central entity helps the subjects to utilize the services. The central entity may provide resource lookup or coordination functionality; the service itself is then established between the subjects without further involvement of the central entity.

Figure 9a visualizes an exemplary centralized P2P system.

DECENTRALIZED P2P SYSTEMS In decentralized P2P systems, no central entity is facilitating the service; all subjects are equally responsible to cooperate to establish the functionality of the service. Resource lookup and coordination are therefore challenging, for example, the question whether a resource is not available or "just not found yet" cannot be easily answered.

Figure 9b visualizes an exemplary decentralized P2P system.

HYBRID P2P SYSTEMS In hybrid P2P systems, the advantages of centralized and decentralized P2P systems are combined. While there are no explicit servers involved, more powerful subjects can be selected as so-called "super-peers". In this role, they coordinate a subset of subjects that are connected to them; for this subset, the respective super-peer acts like a central entity and provides the required coordinative support. The super-peers are interconnected and can, therefore, support each other in facilitating the service functionality.

Figure 9c visualizes an exemplary hybrid P2P system; the blue super-peer subjects coordinate their respective subset of subjects in a centralized fashion and are connected among themselves.

2.2.4 *P2P-/Graph-Topologies*

The connection structure of an unstructured P2P system—or the logical underlay—usually reveals patterns specific to, for example, the application that "provides" the subjects and their connections for the anonymous communication system improved in this thesis.

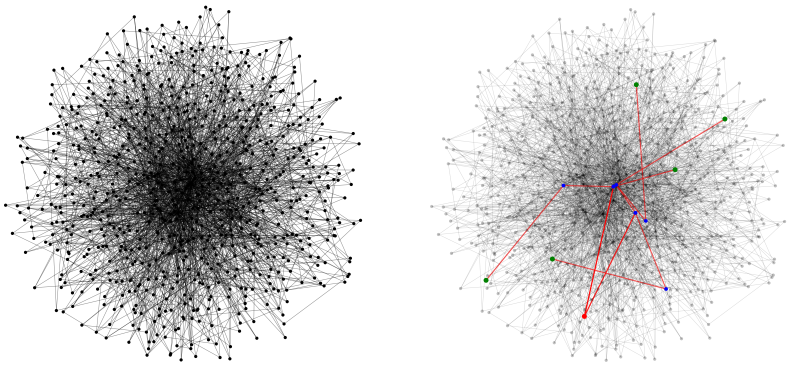
The later evaluation in this thesis bases on the following three models, the social model, the random model, and the random geometric graph (RGG). Additional properties of these networks, beyond the basic ones here, are discussed in Section 4.4.3.

2.2.4.1 *Social Graph*

A social graph results from natural "friend" structures, for example, real-world friendships or Facebook's social graph. In this model, there are many subjects with a comparably low number of connections while a few subjects gain lots of connections. A biased selection probability for choosing destinations for new connections causes a bias towards popular subjects: a higher number of incoming connections result in a higher probability of gaining additional new connections. This biased probability is also called *preferential attachment* and caught by the phrase "the rich get richer" phenomenon in social networks [13, 90, 110].

The highly connected subjects form the so-called *backbone* of the graph and introduce some interesting properties. This backbone often enables a stable—or even an decreasing—(effective) diameter[3] even when the graph is growing, i.e., when additional subjects are joining the system [90]. This property is called *small world* and is usually indicated by an $\mathtt{apl}$ in the order of $\mathtt{ln}(\mathcal{V}$; Cohen and Havlin [25] even suggested that the subjects in the backbone often do not only establish the small-world property

[3] The effective diameters measures the maximal path length that is required to connect a sampled subset of all subjects.



(a) Social Network with 750 subjects

(b) Social Network with 750 subjects and paths from one sender (red) to five recipients (green)

Figure 10: The structure of social networks and the influence of the densely connected backbone on shortest paths.

but the *ultra-small world*, property which is indicated by an apl in the order of $\ln\ln(\mathcal{V})$, when the degree exponent is within $[2,3]$ (see also [12]).

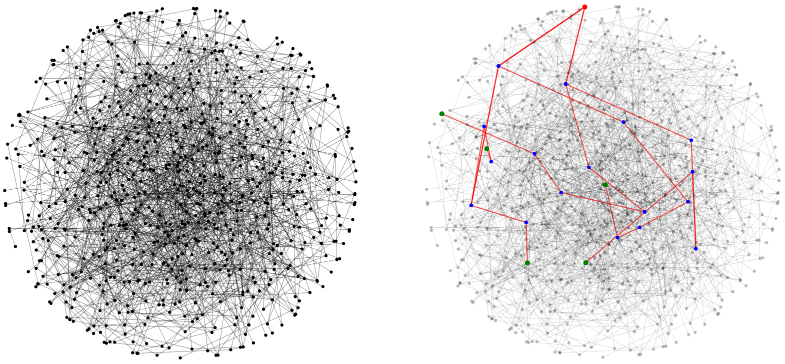
The subjects with a high number of connections, i.e., the subjects in the backbone, provide "shortcuts" through the graph and connect different areas of the graph. These shortcuts also result in a higher load for some subjects in the P2P system as the subjects have to handle messages from a higher number of connections.

The backbone of the system is also visualized in Figure 10a in the darker middle of the network; its influence is visualized by Figure 10b, where a randomly chosen sender (red) is connected to five recipients (green) using shortest paths (for the visualization, usual subjects and unused connections have a lower opacity). All shortest paths are first going into the densely connected center (the backbone) before splitting and leading to their destinations with just a few hops.

2.2.4.2 *Random Graph*

In a random graph, all subjects are equal, and all connections are equally likely such that all subjects the same average degree while the network is not growing. Erdős-Rényi [51] and Gilbert [60] propose two different methods to generate random graphs.

In the Erdős-Rényi model, a random instantiation of $G = (\mathcal{V}, \mathcal{E})$ of all possible permutations is drawn; in contrast to this approach, the Gilbert model evaluates every possible individual connection and does not choose from the set of random graphs. The resulting graphs of both approaches, however, are equivalent.



(a) Random Network with 750 subjects

(b) Random Network with 750 subjects and paths from one sender (red) to five recipients (green)

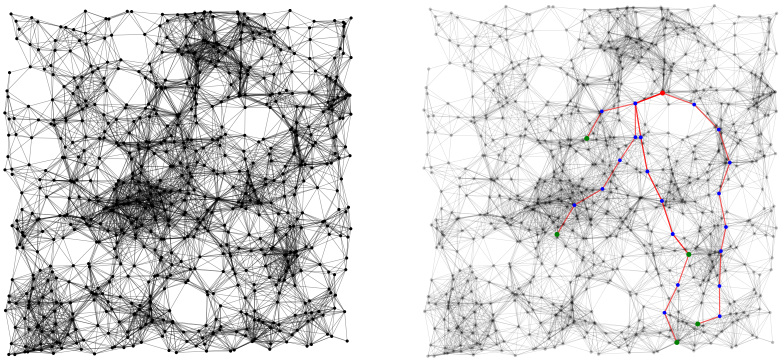
Figure 11: The structure of random networks and the influence of the equality of subjects on shortest paths.

In contrast to the social graphs, random networks are not forming a backbone—all subjects are "equal". As a result of this, the concentration of shortest path in the core of the network is avoided, leading to fairer load distribution in a communication network. Figure 11a visualizes such a random graph and shows no densely connected backbone. Figure 11b shows the influence of the equality of subjects and the missing backbone by visualizing the shortest paths of one sender (red) to five recipients (green). Due to the missing backbone, there is no area of the network where the shortest paths are leading to before splitting to the different recipients.

2.2.4.3 *Random Geometric Graph*

The RGG model—also known as unit disc graph and random euclidean network—assumes subjects that can reach all subjects within a specific distance. An exemplary application scenario of RGGs is a wireless sensor network (WSN) where subjects have the same radio transmission radii, i.e., subjects can reach other subjects that are located within the range of their wireless communication interface. An RGG is established by placing subjects at randomly-chosen coordinates in a d dimensional space.

The pairwise distances of all subjects are computed after placing them in the field; two subjects establish a connection if their distance is smaller than a threshold *t*.



(a) RGG with 750 subjects

(b) RGG with 750 subjects and paths from one sender (red) to five recipients (green)

Figure 12: The structure of RGGs and the influence of distance-based connectivity of subjects on shortest paths.

Figure 12a visualizes an RGG with 750 subjects on a 2-dimensional ($d = 2$, $x \in [0, 1]$, $y \in [0, 1]$) plane with a distance threshold of $t = 0.1$—varying the size of the plane or the threshold t produces sparser (larger plane or smaller threshold) or denser (smaller plane or larger threshold) RGGs. This configuration can be translated into the common "$\|uv\| \leqslant 1$" notation by simple scaling and is "typical" [11, 86, 92, 93] in the research of WSNs and topology adaption. Figure 12b visualizes the shortest paths from one randomly selected sender (red) to five randomly selected recipients (green)—the distance based connectivity does neither yield a densely connected core nor does it foster a specific path pattern.

2.2.5 *Steiner trees*

A Steiner tree is a tree ST of a graph G that spans a set of *terminals* $T \subseteq V$; the costs, or the weight, of a Steiner tree ST are the sum of the costs (or weights) of the connections in ST. The nonterminal vertices in a Steiner tree ST are called *Steiner points*. The *Steiner tree problem* seeks a minimum-cost Steiner tree for a given set of terminals T in a graph G and is NP-complete [79].

Kou et al. [84] provide a heuristic that solves the Steiner tree problem with a distance (approximation factor ρ) of maximal

$2(1 - 1/l)$, with l being the number of leaves in the optimal Steiner tree. The heuristic establishes its Steiner tree approximation (StA) with the following steps, Figure 13 summarizes the heuristic visually:

1. Construct the complete distance graph G_1 from G and T. *The distance graph comprises the terminals in* T*, the connections between them have the costs/weight that correspond the sum of the costs/weight of the connections on the shortest path between the respective terminals.* Figure 13b visualizes the distance graph.

2. Compute the minimal spanning tree, MST_1, of G_1. *If there are multiple minimal spanning trees, select any.* Figure 13c visualizes the minimal spanning tree resulting from the distance graph.

3. Construct the subgraph, G_S, of G by replacing the connections in MST_1 by their shortest path in G. *If there are multiple shortest, select any.* Figure 13d visualizes the subgraph.

4. Compute the minimal spanning tree, MST_2, of G_S. *If there are multiple minimal spanning trees, select any.* Figure 13e visualizes the minimal spanning tree resulting from the subgraph.

5. Construct the StA ST by removing all Steiner points in leaf position iteratively. Figure 13c visualizes the minimal spanning tree resulting from the distance graph.

The example in Figure 13, which is given by Kou et al., uses weighted connections. In the scope of this thesis, the weights of connections are 1, for example, as achieved by using a constant weight function.

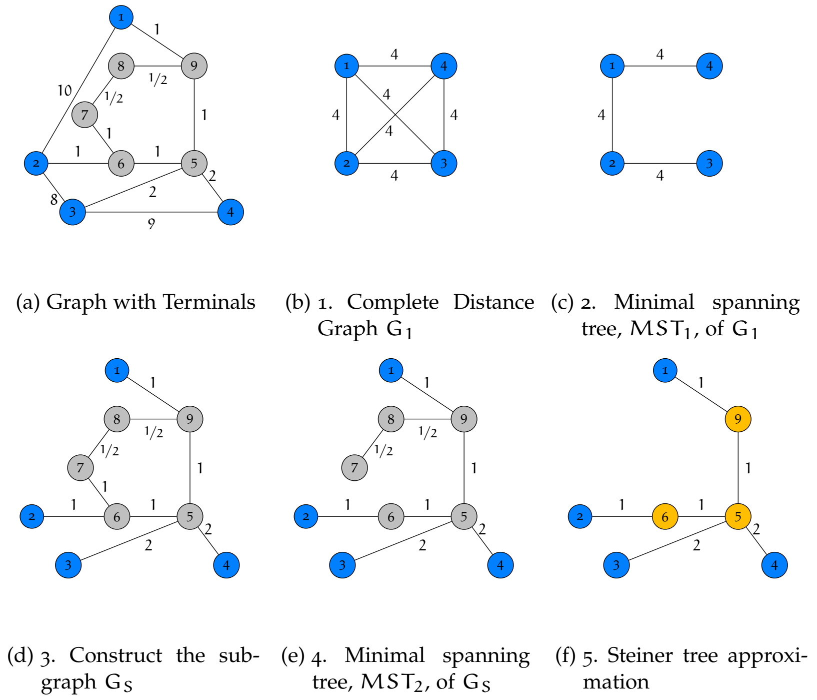
2.2.6 *Hash Functions*

Following Menezes et al. [102], a hash function is a function h that has at least the following two properties:

- *compression:* h maps an input x of arbitrary (but finite) length to an output $h(x)$ of defined and fixed length n.

- *easy computation:* given an input x and a hash function h, the image $h(x)$ is easy to compute.

Additionally, hash functions are *mostly* one-way functions, i.e., while the computation of the image $h(x)$ is "easy" given h and



(a) Graph with Terminals

(b) 1. Complete Distance Graph G_1

(c) 2. Minimal spanning tree, MST_1, of G_1

(d) 3. Construct the subgraph G_S

(e) 4. Minimal spanning tree, MST_2, of G_S

(f) 5. Steiner tree approximation

Figure 13: Computation of a Steiner tree approximation [84]. Vertices in blue are terminals; vertices in yellow are Steiner points.

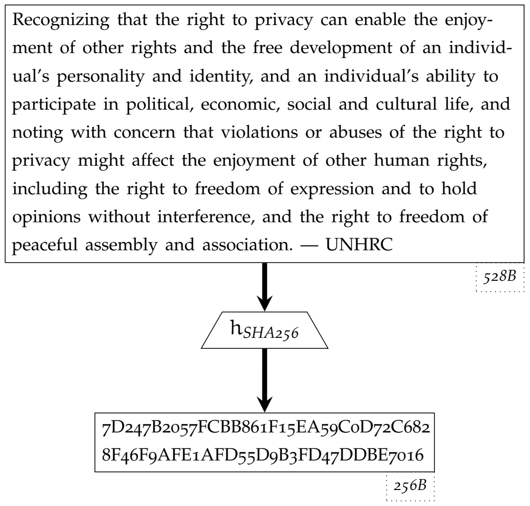
x, the computation of x given h and $h(x)$ is "hard". For easy and hard are no conclusive definitions given. Hard, for example, is typically referred to as *computationally infeasible* [119] or comparable to *solving the factorization problem* [121].

In the scope of this thesis, keyless hash functions are used. These are commonly used for modification detection. Other hash functions are constructed to use a key as second input parameter and are then used, for example, as message authentication codes.

COLLISION RESISTANT HASH FUNCTION Informally, a collision resistant hash function is a hash function h for which the finding of any two inputs x and y that have the same image $h(x) = h(y)$ is "hard" [33, 104, 105, 119, 121]. The collision resistant hash function has to fulfill the following three requirements.

1. *preimage resistance:* given $h(y)$, the image of y under h, it is "hard" to find an input x such that $h(x) = h(y)$.

2. *2nd-preimage resistance:* given x, it is "hard" to find a second input y which has the same image such that $x \neq y$ and $h(x) = h(y)$.



Recognizing that the right to privacy can enable the enjoyment of other rights and the free development of an individual's personality and identity, and an individual's ability to participate in political, economic, social and cultural life, and noting with concern that violations or abuses of the right to privacy might affect the enjoyment of other human rights, including the right to freedom of expression and to hold opinions without interference, and the right to freedom of peaceful assembly and association. — UNHRC

7D247B2057FCBB861F15EA59C0D72C682
8F46F9AFE1AFD55D9B3FD47DDBE7016

Figure 14: SHA-256 hash function example. A 528 byte text-input (the UNHCR-quote from Chapter 1) is mapped to its 256 byte image.

3. *collision resistance:* it is "hard" to find any two (different) inputs x, y such that they have the same image under h, i.e., $h(x) = h(y)$.

2.2.7 *Bloom Filter*

Bloom filters [16, 57] $\mathbf{bf}_i$ are containers that store information in an array and (mainly) support two operations, the *adding* of elements and the *contains*-check.

As such, Bloom filters enable to quickly check whether an element has been "processed" before without involving possibly expensive disk or network lookups for non-existent information.

A Bloom filter holds its elements in an array of size m_{bf} which is initialized with zeros. An element is added by computing its hash value under k_{bf} (different) hash functions and setting the respective entries of the vector to one.

By this construction, the Bloom filter can provide a definite answer for the contains operation, if at least one of the k_{bf} entries is zero. Then, the element is undoubtedly not contained. If all k_{bf} entries contain ones, the Bloom filter *may* contain the element—an indisputable positive response cannot be given, as the k_{bf} fields may have been set by the adding of other elements (i.e., the contains-check may return true through collision).

Deleting elements from a (simple) Bloom filter is not possible as the fields may also be set to one by the addition of another element. Then, the deletion of an element would introduce false negatives to the Bloom filter. Fan et al. [57] introduced *counting* Bloom filters as an improvement to the classical Bloom filters. Here, the entries are not set to one but used as a counter—the Bloom filter increments the counter every time an entry is used to add an element; accordingly, the Bloom filter decrements the respective counters when removing respective elements. As a result of using these counters, the Bloom filter can remove elements without introducing false negatives.

> **Example 14:** Figure 15 visualizes an exemplary Bloom filter **bf**. **bf** consists of a vector with $m_{bf} = 20$ fields and is filled by $k_{bf} = 3$ hash functions. Six elements are added to the **bf**—assume that element 6 is not added to **bf**, the contains operation will still return a positive response as the contained elements 1, 2, 4, and 5 set the respective entries of element 6 to one.

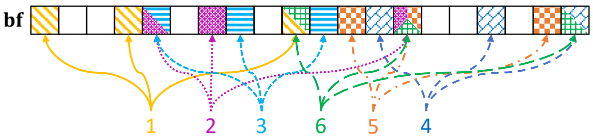


Figure 15: A Bloom filter **bf** with a $m_{bf}=20$ bit vector, $k_{bf}=3$ hash functions, and six pieces of information added (1–6).

CONFIGURING THE BLOOM FILTER TO ACHIEVE A FALSE POSITIVE PROBABILITY A Bloom filter returns false positives with the probability p_{fp} which depends on the number of inserted elements n and the configuration of the Bloom filter **bf** which is defined by the size of the array m_{bf} and the number of hash functions k_{bf}.

Equation (3) formulates the probability that a particular entry of the Bloom filter is zero after the insertion of n elements.

$$\left(1 - \frac{1}{m}\right)^{kn} \tag{3}$$

Using Equation (3), the probability of a false positive evaluation can be computed with Equation (4).

$$p_{fp} \approx \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)^{k} \leqslant (1 - e^{mk/n})^{k} \tag{4}$$

The false positive probability can be pre-computed by estimating the number of elements that the Bloom filter will store. These precomputed probabilities can then be used to configure the Bloom filter **bf** according to the requirements of its application, for example, demanding a limited number of false positives. Appendix C (page 227 ff.) provides tables with pre-computed false positive probabilities for variations of k_{bf}, m_{bf} and n.

MERGING BLOOM FILTERS Bloom filters can be merged by combining their respective vectors with bit-wise application of the OR-function as described by Equation 5. With that, the contents are preserved and combined in a single Bloom filter of the same size.

$$\forall i \in \{1, 2, \dots, m_{bf}\}: \ \mathbf{bf_m}[i] = \mathbf{bf_l}[i] \ || \ \mathbf{bf_r}[i] \qquad (5)$$

The size of the all merged Bloom filters, for example, $\mathbf{bf_l}$ and $\mathbf{bf_r}$, as well as the size of the resulting Bloom filter $\mathbf{bf_m}$ stays stable. With that, the false positive probability can increase considerably. Example 15 shows the considerable change of the false positive probability when merging two Bloom filters.

> **Example 15:** Figure 16 visualizes this process by merging the Bloom filters $\mathbf{bf_l}$ and $\mathbf{bf_r}$. The entries 7 and 8 of Bloom filter $\mathbf{bf_l}$ as well as the entries 6 and 9 of Bloom filter $\mathbf{bf_r}$ are preserved in the resulting Bloom filter $\mathbf{bf_m}$.
> The respective false positive probabilities are:
>
> $$\begin{aligned} P_{fp,\mathbf{bf_l}} &= 0.0174 \quad (m_{bf_l} = 20, k_{bf_l} = 3, n = 2) \\ P_{fp,\mathbf{bf_r}} &= 0.0174 \quad (m_{bf_r} = 20, k_{bf_r} = 3, n = 2) \\ P_{fp,\mathbf{bf_m}} &= 0.0918 \quad (m_{bf_l} = 20, k_{bf_l} = 3, n = 4) \end{aligned}$$

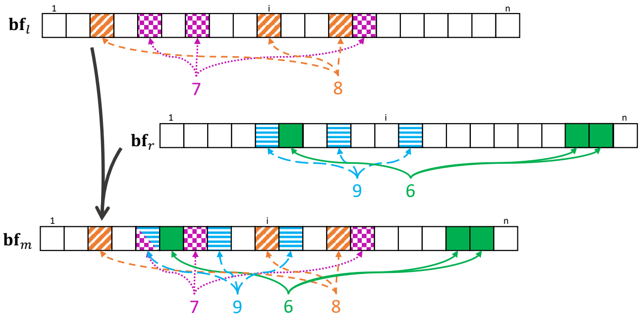


Figure 16: Merging two Bloom filters $\mathbf{bf_l}$ and $\mathbf{bf_r}$.

2.3 ADVERSARY MODEL

Subjects in communication systems face severe challenges regarding information leakage. This section introduces the properties and capabilities, which will then be founded by real world manifestations of adversaries. After that, attacks on users' anonymity are introduced and discussed.

2.3.1 *Properties and Capabilities*

The achieved and achievable anonymity highly depends on the faced adversary $\mathfrak{A}$. The adversary is defined by a *goal*, their *interaction* capabilities, and their *locality* restrictions.

GOAL In the scenario of anonymous communication systems, the adversary tries to break either anonymity or confidentiality, or both. Breaking anonymity allows an adversary to link a message to its sender and recipients. Breaking confidentiality allows an adversary to learn about the content of a message.

As this thesis is dedicated to improving anonymous communication rather than confidential communication, the adversary follows the goal of breaking anonymity.

INTERACTION An adversary $\mathfrak{A}$ can be distinguished by their power to interact with other subjects. The adversary can, however, only interact with messages according to their local spread, i.e., the adversary can interact at all subjects $c \in \mathfrak{C}$ that they control. (see LOCALITY below).

An adversary can be limited to be *passively* observing. Such an adversary $\mathfrak{A}_{p,\_}$ will follow the protocol strictly, i.e., the adversary will handle messages as defined by the communication system. However, the adversary $\mathfrak{A}_{p,\_}$ will gather and learn every information that is leaked by messages that are accessible by the adversary.

The adversary $\mathfrak{A}_{p,\_}$ is restricted to the following functions:

- `handleMessage()`: process a message as required by protocol.

  $\mathfrak{A}_{p,\_}$ *learns every information from the message but will not violate the protocol otherwise.*

In contrast to such a passive adversary, an adversary $\mathfrak{A}_{a,\_}$ can also be *actively* interfering with the communication. That is, they can, for example, replay, drop, delay, messages that they see at subjects under their control or connections respectively. Moreover, as the adversary $\mathfrak{A}_{a,\_}$ possesses the cryptographic key material, they can create and emit new (and valid) messages. The adversary $\mathfrak{A}_{p,\_}$ is unrestricted in their interaction; the adversary $\mathfrak{A}_{p,\_}$ can, among others, therefore use:

- `handleMessage()`: process a message as required by protocol.

  $\mathfrak{A}_{a,\_}$ *learns every information from the message but will not violate the protocol otherwise.*

- `createMessage()`: create a new message; the message is not distinguishable from an authentic message.

  $\mathfrak{A}_{a,\_}$ *may learn from potential replies, delays, and the visible message handling.*

- `replayMessage()`: emit a previously recorded message again into the system.

  $\mathfrak{A}_{a,\_}$ *may learn from potential replies, delays, and the visible message handling.*

- `dropMessage()/delayMessage()`: drops or delays a message before processing the message according to the protocol.

  $\mathfrak{A}_{a,\_}$ *may learn from reactions of other subjects like re-send messages.*

LOCALITY An adversary $\mathfrak{A}$ is restricted in applying their capabilities to their local spread, i.e., an adversary can only interact (see INTERACTION above) with resources that are accessible to them. The set of subject that are under the control of the adversary $\mathfrak{A}$ is denoted by $\mathfrak{C}$.

- The adversary $\mathfrak{A}_{\_,l}$ can be *limited* to controlling a single subject c in the network $|\mathfrak{C}| = 1$, i.e., they can interact with and learn from this single subject's point of view and its incident connections:

$$\{e_i : (\forall c \in \mathfrak{C})(\forall \nu_k \in N(c)) \\ [e_i = (c, \nu_k) \in \mathcal{E} \ \vee \ e_i = (\nu_k, c) \in \mathcal{E}]\}$$

- A *colluding adversary* $\mathfrak{A}_{\_,c}$ controls and coordinates multiple subjects $c_i \in \mathfrak{C}$ with $|\mathfrak{C}| > 1$. However, it is a realistic assumption that a non-state attacker is bound in the number of controlled subjects $|\mathcal{V} \setminus \mathfrak{C}| > 2$, that is, there are *some* (at least 1) non-hostile subjects that aim at enabling anonymous communication. As a result of this, the adversary $\mathfrak{A}_{\_,c}$ can learn from its controlled subjects $c_i \in \mathfrak{C}$ and their incident connections:

$$\begin{aligned}&\{e_i : (\forall c_i \in \mathfrak{C})(\forall v_k \in N(c_i))\\ &\qquad [e_i = (c_i, v_k) \in \mathcal{E} \vee e_i = (v_k, c_i) \in \mathcal{E}]\}\end{aligned}$$

- A *global adversary* (or *state-level* adversary) $\mathfrak{A}_{\_,g}$ is aware of the whole network, e.g., by having access to the connections themselves. Thus, a global adversary is hard to defend against; nonetheless, such an adversary should be considered as an extreme case. The adversary $\mathfrak{A}_{\_,g}$ can learn from all connections:

$$\{e \in \mathcal{E}\}$$

The adversary $\mathfrak{A}$ is then defined as a combination of interaction and locality:

$$\begin{aligned}&\mathfrak{A} \in \{\mathfrak{A}_{p,\_}; \mathfrak{A}_{a,\_}\} \times \{\mathfrak{A}_{\_,l}; \mathfrak{A}_{\_,c}; \mathfrak{A}_{\_,g}\}\\ &\quad \Rightarrow \mathfrak{A} \in \{\mathfrak{A}_{p,l}; \mathfrak{A}_{p,c}; \mathfrak{A}_{p,g}; \mathfrak{A}_{a,l}; \mathfrak{A}_{a,c}; \mathfrak{A}_{a,g}\}\end{aligned}$$

2.3.2 *Real World Manifestations*

The last two decades revealed adversary manifestations that are realistic, especially the last few years have shown that adversaries are stronger than expected. The increase of power of realistic adversary manifestation is, among others, caused by the increase in available data. This thesis assumes that these powerful adversary models are realistic and should be used to provide a worst-case analysis.

COLLUDING OR NEAR GLOBAL, PASSIVE ADVERSARY A service provider is a near global adversary as they can access nearly all emerging data. Despite the theoretical ability of active interference, the service provider will most likely be passively observing to avoid risking their business model.

> **Example 16:** Bob, the service provider, represents a near global adversary that can passively observe the communication of the subjects participating in the communication service. By being able to observe all communication, the service provider can break anonymity.

The *curious service provider* is shown to be realistic by, for example, Uber's[4] "rides of glory" [144], de Montjoye et al. [41], and Narayanan and Shmatikov [109].

GLOBAL, PASSIVE OR ACTIVE ADVERSARY A state-agency is a global adversary being able to connect data from different services. Such an adversary is not limited in any way and can access data without restriction concerning their location. Despite being able to interfere with the communication actively, laws usually restrict power of adversaries to passive: while a service provider may gain the right to access and analyze data using their terms (thus, circumventing for example StGB §§201,202a–c), actively interfering with the communication and emerging data remains questionable (see, for example, StGB §§206(2)2.,303a-b). Laws, however, are a fragile construct to argue for limiting the power of adversaries as they may be adapted in the adversary's favor—this adaptation of laws may happen before such an adversary is implemented, for example, by implementing the so-called

> *Gesetz zur Beschränkung des Brief-, Post- und Fernmeldegeheimnisses (Artikel 10-Gesetz - G 10) ["Act on Restrictions on the Secrecy of Mail, Post and Telecommunications, G10 Act"] [21],*

and the

> *Foreign Intelligence Surveillance Act of 1978 "FISA" [26] which got several amendments as, for example, the USA PATRIOT Act and the USA FREEDOM Act [27, 28].*

In both cases, laws have been adapted to grant adversaries (here: intelligence agencies) additional rights. It is also possible that the laws framing the power of the adversary are adapted in retrospective [100, 147].

> **Example 17:** A state-agency, for example, intelligence, accesses and observes communication of Sally and Rey in Bob's service by tapping into the connections of Sally, Rey, and Bob itself or lawfully requesting the data from Bob directly. By being able to access these data, the adversary is able to break anonymity.

[4] ride-sharing service Uber: https://www.uber.com

This type of global adversary is shown to be realistic by EU reports about the ECHELON surveillance program [115] and the so-called Snowden revelations [65, 97].

The combination of INTERACTION and LOCALITY of beforementioned exemplary real world manifestations are summarized in Table 1.

Table 1: Combinations of Interaction and Locality of Real World Adversary Models

	Interaction		Locality			
	active	passive	local	colluding	global	
curious service prov.	×	✓	×	✓	✓	$\mathfrak{A}_{p,c}$, $\mathfrak{A}_{p,g}$
intelligence agency	✓	✓	×	×	✓	$\mathfrak{A}_{p,g}$, $\mathfrak{A}_{a,g}$

2.4 ANONYMOUS COMMUNICATION

Anonymous communication—and the enabling anonymous communication systems—is strongly dependent on primitives that provide anonymization and metrics that measure the quality of anonymity, i.e., measure "how much" privacy is being achieved by the communication system. This section introduces common and prevailing anonymity metrics. After that, this section introduces and discusses the primitives that are used to provide anonymization in most anonymous communication systems under the angle of efficiency and anonymity in a group communication scenario.

2.4.1 *Anonymity Metrics*

Anonymity measurement is a much-debated field of research. A rigorous measurement of anonymity is necessary to reply to the ever-present question of "how much anonymity does the communication system achieve?"

Many anonymity metrics, for example, k-Anonymity, l-Diversity, t-Closeness, were initially developed to measure anonymity in data sets but are today also used to measure anonymity in communication systems.

This section summarizes the prevailing metrics to measure anonymity.

ANONYMITY SET SIZE The anonymity set size measures anonymity as the number of subjects that are indistinguishable from each other. According to [116], the anonymity set size (ass) measures the minimal number of subjects that form a group which hides the subjects (either senders s or recipients r).

A similar measure of anonymity is commonly used in DCnets (for example, [22, 30, 61]), Mix-nets (for example, [15, 23, 24]), and also in k-Anonymity [128].

K-ANONYMITY AND ITS SUCCESSORS Samarati and Sweeny proposed *k-Anonymity* [128, 139] as a metric to measure anonymity and anonymize in datasets that are to be published. According to Samarati and Sweeny, a dataset is k-anonymous, if each subject in an equivalence class yields at least $k-1$ other subjects. k is an adjustable parameter and defines the size of the anonymity set. k-Anonymity is, among others, susceptible for lack of diversity in the sensitive attributes and adversaries with additional background information.

Machanavajjhala et al. [98] proposed a measure called *l-Diversity* to overcome the weaknesses of k-Anonymity. l-Diversity requires each equivalence class to contain at least l different sensitive attributes. As a result of this, l-Diversity ensures that an adversary may guess the sensitive attribute with a confidence of maximum $^1/_l$.

However, the degree of sensitivity for sensitive attributes is not equal for all of these attributes. Thus, even l-diverse datasets may leak information which can break anonymity. Li et al. [91] proposed *t-Closeness* as a measure to ensure that sensitive attributes themselves do not leak information. t-Closeness requires the sensitive attributes in an equivalence class to have a distribution of values that are at a maximum distance of t from the distribution of values in the original full dataset.

Xiao and Tao [150] pointed out that datasets are often republished with changes (inserted and removed entries) while k-Anonymity, l-Diversity, and t-Closeness assume the dataset to be anonymized and published once. Republication with updated values yields the possibility of statistical inference by analyzing and correlating consecutively released datasets. Xiao and Tao proposed *m-Invariance* to overcome this challenge by ensuring that at least m entries are invariant in consecutive anonymized datasets.

entropy-based anonymity Set-based anonymity metrics assume that the probability of actually being a sender or recipient is equal for all participants populating the respective anonymity sets. Adversaries $\mathfrak{A}$ are often able to utilize additional knowledge during their attacks. Thus, the assumption of equal probability does not hold in many cases.

> **Example 18:** Assume an anonymous communication system that establishes communication overlays O_t based on shortest path. O_t will then be a dissemination tree with the sender being the root. If the adversary $\mathfrak{A}$ is able to utilize topological knowledge in their attack, they will realize that leaf-subjects (that are subjects without outgoing connections) have a probability of $p_{leaf} = 1$ of being a recipient. The remaining "inner" subjects have a probability of $p_{inner} = (|\mathcal{R}_t| - |leaf-subjects|)/(|\mathcal{V}_t| - |leaf-subjects|)$, thus, $p_{inner} \leqslant p_{leaf}$.

Serjantov and Danezis [130] and Diaz et al. [44] propose models to measure anonymity using the Shannon entropy [131]. Anonymity is measured by computing the probability of an adversary $\mathfrak{A}$ to identify a subject in the whole population $\mathcal{V}$ (in related work often denoted as Ψ) as being part of the group of *active participants* $\mathcal{S}_t \cup \mathcal{R}_t$. For that, they establish a probability distribution p that models the posterior probability p_u of the adversary $\mathfrak{A}$ identifying subjects $u \in \mathcal{V}$ as being a sender or recipient. The Shannon entropy over p (given in Equation (6)) evaluates the achieved anonymity by measuring the certainty of the adversary in identifying the active participants.

$$S = -\sum_{u \in \mathcal{V}} p_u \log_2(p_u) \tag{6}$$

Diaz et al. normalize the Shannon entropy (Equation (6)) to provide a normalized *anonymity degree* as visualized in Equation (7). The maximum entropy provided by $S_{max} = \log_2(|\mathcal{V}|)$. Diaz et al. assume a degree of 0.8 to be sufficient to provide valuable anonymity. However, they suggest that the minimum acceptable anonymity degree depends on anonymity requirements of the actual system which in turn require intensive model testing.

$$d = 1 - \frac{S_{max} - S}{S_{max}} = \frac{S}{S_{max}} = \frac{S}{\log_2(|\mathcal{V}|)} \tag{7}$$

selecting the appropriate metric The measurement of anonymity depends on system properties and the selection of an appropriate adversary model. Recently, Danezis [35] pointed out that the answer to the question of anonymity measurement is not answered satisfactory. Nonetheless, users of the system need to understand the applied anonymity metrics. Thus, the selection of the "correct" anonymity metric depends on required level and granularity of anonymity measurement by the communication system, and the ability of the audience to understand the anonymity evaluation.

The contributions of this thesis are evaluated with both anonymity set size and the entropy-based anonymity degree. The more intuitive anonymity set size is used where appropriate; the entropty-based anonymity degree is used where the anonymity set size fails to provide an accurate anonymity measurement.

2.4.2 *Anonymization Primitives*

Most anonymous communication systems utilize one (or multiple) of a set of anonymization primitives. These primitives were typically developed during the 1980s and 1990s. This section provides an overview of these primitives.

2.4.2.1 *Proxy*

Proxy-based anonymization decouples senders s_i and recipients r_j by inserting an intermediate entity ("proxy") between them. This intermediate entity is the proxy or broker that is relaying messages between sender s_i and recipient r_j.

While the proxy in between s_i and r_j allows senders and recipients to stay anonymous concerning each other, the proxy itself can identify and link sender and recipient(s) of a specific instance of communication.

> **Example 19:** Sally sends a message to Rey using Bob as a broker. Using Bob, Sally can stay hidden from Rey. However, Bob can easily link the message to Sally and Rey.

2.4.2.2 *Mix Network*

Chaum [23] proposed mix networks as "untraceable, anonymous remailers"—an asynchronous anonymization system to exchange messages anonymously. Mix networks extend the idea of proxy-based anonymization with

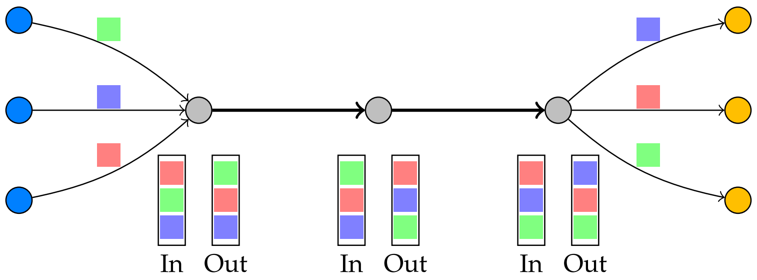


Figure 17: Anonymization Primitive: Mix network.

- collecting and shuffling of messages, and
- chaining of multiple of these collecting and shuffling subjects (called *mix nodes*).

The shuffling follows a *random permutation* of the previously collected messages. Together with the multi-hop forwarding. There is no single broker able to identify sender and recipient at the same time and lessens the chances of a global adversary to identify sender and recipient as well. Figure 17 shows a mix chain of three mix nodes, a full mix network will be composed of multiples of those chains that are either fixed by the service provider (*cascades*) or selected by the subjects (*free routes*). In Figure 17, every mix node receives a set of messages, depicted by the colored squares, shuffles them and relays them to the next mix node; the repeated shuffling in combination with the changing encryption in every hop prevents an adversary $\mathfrak{A}_{\_,g}$ from tracing communication messages from sender to recipient.

2.4.2.3 *Onion Routing*

Goldschlag et al. [62, 124] introduced Onion Routing (OR) as an improvement of mix networks. OR lowers the burden of performing the actual permutation of mix networks; willing subjects can join the system as node to support the anonymization. More precisely, anybody willing to contribute can set-up a computer to act as a relay. Subjects can select their chains of relays (called *circuits*) depending on their preferences and establish their communication using their respective circuit.

Figure 18 visualizes such an exemplary circuit. The blue sender adds all layers of encryption and sends the message to the first subject in the circuit. This subjects removes one layer of encryption and forwards to the next subject in the circuit which is now visible. The following subject also removes one layer of encryption and forwards the message to the last subject in the circuit; this last subject removes the last layer of encryption and delivers the message to the recipient.

To improve the latency of mix networks, OR removes the batching and shuffling of messages, and forwards messages immediately to the next hop.

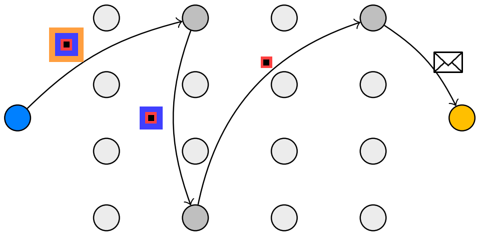


Figure 18: Anonymization Primitive: Onion Routing (OR).

2.4.2.4 *Randomized Forwarding*

Reiter et al. [125] proposed *randomized forwarding* which hides communication relationships by using a distributed broker network. Subjects work together as brokers to deliver the message to its recipients. By probabilistically relaying messages until a random broker delivers the message to its destination, a non-global adversary cannot easily link a message to a sender and recipient.

Due to the probabilistic behavior, messages circle among the subjects (the broker network) for an unknown and undefined amount of time before the messages arrive at their destination.

Figure 19 visualizes the exemplary dissemination of three consecutive messages towards a single recipient; these messages use potentially different paths as every subject that processes a messages decides independent from other subjects whether to deliver or relay (and to which subject to relay).

2.4.2.5 *Cover Traffic*

Cover traffic, as used, for example, in [15, 58], hides communication messages behind noise, i.e., hides communication messages behind messages with random content. The noise is indistinguishable from real communication messages, for example, by applying encryption; the indistinguishability of communication messages and noise prevents adversaries from learning implicit

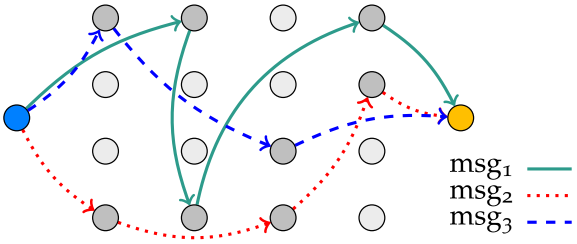


Figure 19: Anonymization Primitive: Randomized forwarding.

metadata. A global adversary is only able to identify *active* connections; thus, the subjects are required to utilize each connection in each round. The adversary can perform an intersection attack otherwise, effectively removing the noise and identifying senders and receivers over time.

Figure 20 visualizes partial cover traffic. The blue sender sends their message; the cover subjects in dark-gray are sending a cover message. From an adversary's perspective, the messages are indistinguishable. As a result of this, the message flow is no longer visible and the adversary cannot collect evidence about senders and recipients.

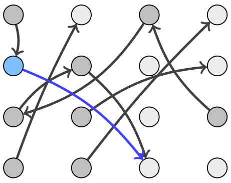


Figure 20: Anonymization Primitive: Cover traffic.

2.4.2.6 *Dining-Cryptographer Network*

Chaum [22] introduced the Dining-Cryptographer (DC) problem. In DCnets, subjects send their messages to all other subjects in the system. Based on secure multi-party computation, every subject in the DCnet has to contribute an encrypted message. These messages are then combined to derive the sender's message. This combination can be performed by every subject; thus, all subjects are potential recipients.

The message of the sender can provably not be linked to any of the subjects; the DCnet provides anonymity in the set of honest participants of the DCnet.

Figure 21 visualizes a small DCnet where five subjects are sharing their part of the multi-party computation. As a result of this multi-party computation, all subjects are aware of the message that the blue-orange sender s included in their part of the multi-party computation.

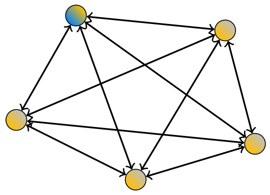


Figure 21: Anonymization Primitive: Dining-Cryptographer network.

2.5 SUMMARY

This chapter introduced definitions, basic terminology, and model, as well as metrics and anonymization primitives. This information is essential to understand and motivate the mechanisms presented in this thesis. Table A.26 provides a summarizing overview of the introduced terminology and abbreviations.

A distinction of anonymity, confidentiality, and privacy has been given. This distinction is followed by a detailed reflection of the different forms and levels of anonymity. Anonymity manifests itself in four stages with increasing protection: unlinkability, k-anonymity, absolute anonymity, and untraceability.

Next, this chapter introduced a communication model based on the Pub/Sub paradigm. This communication model can represent group communication and anonymization techniques and introduces senders and recipients as active participants (subjects) of a communication, brokers as enabling subjects as well as overlays as a level of abstraction. The introduction of Pub/Sub-based communication for P2P systems is the communication model for the later chapters. A differentiation of different classes of P2P systems, namely unstructured and structured P2P systems, complements the P2P communication model.

After fixing the model and terminology, this chapter discusses metrics to measure anonymity. The introduced metrics can be divided into two big classes: set-based and entropy-based metrics. The set-based anonymity measures express anonymity by evaluating the size of the group of indistinguishable subjects in which senders and recipients are hiding. Entropy-based anonymity measures express anonymity as the certainty of an adversary about active roles of subjects in a communication.

Following these definitions, this chapter explains the core requirements of efficiency and anonymity and introduces auxiliary requirements that are essential to provide efficient and anonymous group communication at large scale.

3 STATE OF THE ART

This chapter introduces characteristics of anonymous communication systems and discusses and compares state of the art in anonymous communication systems according to these characteristics. The chapter is structured as follows: Section 3.1 introduces requirements for anonymous communication systems; Section 3.2 expounds system characteristics that influence efficiency, anonymity, and structure of anonymous communication systems—properties that show challenges that need to be focused. Section 3.3 then discusses and compares state of the art systems that aim to provide anonymous communication. The discussion is structured according to the main anonymization primitive used in the respective system. In Section 3.4, the current state of the art is recapitulated, and conclusions are drawn.

3.1 REQUIREMENTS

This section provides an introduction to requirements that are relevant for contributions of this thesis. First, this section introduces the two central requirements of this thesis efficiency and anonymity. Second, this section introduces a selection of secondary requirements. These secondary requirements are important for the functionality of anonymous communication systems and noted for completeness. However, as this thesis focusses on efficiency and anonymity, the secondary requirements are not discussed in detail.

3.1.1 *Efficiency*

Efficiency is a core requirement to achieve widespread usage of a communication system, which is apparent when comparing the adoption of anonymization services as, for example, Mixminion [34] and Tor [46]. Tor in contrast to Mixminion considered the time-efficiency important; as a result of this, Tor was able to become one of the leading anonymization services. As described in Section 2.1.1, efficiency is not only considering time behavior but also message load.

Efficiency requires to *lowering the number of transmitted messages* across the system to unburden the subjects in the system. Also, the connection stress [135], i.e., the number of duplicate messages over a connection, should be minimized.

In the scope of this thesis, efficiency abstracts from computational overhead, and energy and memory consumption.

3.1.2 *Anonymity*

Anonymity is the ability of the communication system to hide senders and recipients, as well as their respective relationships.

Following the definitions given in Sections 2.1.2 and 2.1.3, anonymity is provided if at least unlinkability with level-3 is given. For that, adversary according to Section 2.3 are considered for an anonymity analysis based on graph-theoretical overlay properties that are induced by the establishment of the communication overlay.

3.1.3 *Secondary Requirements*

Auxiliary requirements cover those requirements that are important to the overall functionality. These requirements are, however, only discussed in brief to show that they are included in considerations and still fulfilled but not evaluated in all detail.

3.1.3.1 *Confidentiality*

Confidentiality is the ability of the system, to hide the contents of a communication from unauthorized subjects, i.e., the system discloses the content of messages only to intended subjects.

Confidentiality in a communication system based on Pub/Sub comes in two flavors:

- *Information Confidentiality* is concerned with the protection of the content of the communication itself. Information confidentiality ensures that unauthorized are not able to learn anything from the messages by handling them.

- *Signaling Confidentiality* is concerned with the hiding of the content of signaling messages (advertisements and subscriptions). Thus, signaling confidentiality ensures that unauthorized subjects are not able to learn about the topics of advertisements and subscriptions from messages that they handle.

3.1.3.2 *Accountability*

Accountability requires that all events (sending and receiving a message) can be associated with a subject. This association of events to subjects is not to be impeachable, i.e., if an event is associated with a subject there has to be substantial evidence.

By requiring the ability to attribute events to subjects, accountability is in tension with anonymity.

3.1.3.3 *Integrity*

Integrity ensures correctness and completeness of communication, i.e., integrity provides evidence that messages are neither altered nor suppressed. According to Wang et al. [148], integrity has three flavors in Pub/Sub-based systems:

- *Information Integrity* is concerned with the messages. Information integrity ensures that a message is not altered and originates indeed from the sender.

- *Subscription Integrity* ensures that an adversary cannot alter the interests of recipients without authorization. These interests are essential to enable routing and message forwarding.

- *Service Integrity* ensures that the anonymous communication system itself is not maliciously acting, i.e., the service itself does not behave outside of its parameters. With this, the service performs all functionality as agreed.

3.1.3.4 *Availability*

Availability requires that the system be (functionally) available to subjects whenever they request the service. That includes the ability to advertise topics, subscribe to topics and to send and receive information. Also, availability requires successful routing of messages, i.e., requires a successful and complete establishment of communication overlays to ensure that all recipients can connect to the respective senders and vice-versa.

Counterexamples may be induced by, for example, denial of service (DoS) attacks or insufficient scalability (see Section 3.1.3.5). Using DoS attacks, an adversary tries to paralyze the service by exhausting the resources of the service, for example, by overloading the communication system with requests. Insufficient scalability can diminish the availability of the service, for example, in the expensive overlay establishment phase where subjects exchange plenty of messages.

As requirement, availability is tightly coupled to efficiency (Section 3.1.1) and scalability (Section 3.1.3.5).

3.1.3.5 *Scalability*

Scalability describes the ability of a system to cope with increasing load, for example, by an increasing number of subjects or increasing number of messages that subjects exchange.

Scalability may be limited by complex routing mechanisms and expensive optimizations. For example, reliability guarantees enforce extensive logging to enable detection and retransmission of lost messages; thus, these reliability guarantees require lots of memory at each subject which in turn decreases scalability due to resource limitations.

Limited scalability can affect availability (Section 3.1.3.4) if the subjects are not able to cope with increasing load.

3.2 CHARACTERISTICS

Anonymous communication systems exhibit characteristics that affect their ability to achieve efficient anonymous group communication positively—or negatively. In this section, a set of characteristics is introduced that is used to compare the state-of-the-art in the field of anonymous communication systems with each other.

3.2.1 *Communication Paradigm*

Selecting the appropriate communication paradigm for an application scenario is essential to reduce the number of identical messages in the communication system.

Naturally, there are four communication paradigms: *one-to-one* (1-to-1), *one-to-many* (1-to-n), *many-to-one* (m-to-1), and *many-to-many* (m-to-n). When analyzing efficiency and anonymity a single message is focussed; therefore, this thesis uses only the *one-to-one* (1-to-1), *one-to-many* (1-to-n) paradigms. The latter two paradigms (m-to-1 and m-to-n) can be represented by repeating the first two paradigms (1-to-1 and 1-to-n) m-times with m different senders. Therefore, neglecting the latter two communication paradigms does not involve a loss of generality.

As already described in Section 2.1.1, late message duplication is a key feature of native one-to-many communication. As such, late message duplication reduces the message overhead in most application scenarios drastically.

Despite it's being hardly realized in today's communication systems, native group communication is essential for communication in heterogeneous and distributed environments. Many communication services that offer group communication to their subjects, e.g., WhatsApp [111, 112], are realizing the group communication essentially using repeated unicast transmissions (with some server-based performance optimization) to realize group communication.

> **Summary considering efficiency and anonymity:**
> The load in an P2P system can be reduced by duplicating messages as late as possible. As such, selecting the appropriate communication paradigm influence efficiency by controlling the number of messages that need to be handled by the system.
> The communication paradigm alone does not influence the anonymity.

3.2.2 *Connection Mode*

A communication system can be characterized by its implemented connection mode. A *connection-oriented* communication system establishes a connection spanning all involved subjects; this connection is then used for usually both directions and for more than a single communication event (assuming that the communication lasts for more than a single message).

The second method is to establish communication on a per-message basis, using a *connectionless* message transmission. This connectionless approach establishes multiple overlays for each direction of communication, each of which is then optimized for its direction; moreover, it also allows to utilize different paths for each communication message.

Both connection modes have their advantages with respect to anonymity. A connection-oriented system design allows the communication system to adjust protective measures to the needs of this very connection. Yet, using a connection repeatedly increases the success probability of adversaries performing traffic analysis attacks (e.g., the statistical disclosure attack [36]) as the number of potential communication partners can be reduced with every communication event. A connectionless approach avoids the establishment of connections, establishing only a loosely coupled overlay. In such a communication overlay, multiple messages from the same sender to the same recipient may be routed along different paths, thus, increasing the number of potential communication participants.

> **Summary considering efficiency and anonymity:**
> The connection mode itself does not influence the efficiency; both connection modes can transmit communication messages with the same efficiency.
> The connection mode influences anonymity. A connection-oriented system establishes connections that are used for multiple messages. As a result of this, anonymity protection measures can be applied. In contrast to that, the connectionless system may use different routes for every message. As a result of this, an adversary may not be able to observe all messages—which resembles the randomized forwarding primitive.

3.2.3 *Topology*

The topology of a communication system represents the structure of connections that are linking the subjects and, thus, are used to establish the communication. Here, three classes of topologies are considered: *centralized*, *hybrid*, and *decentralized* topologies (Figure 9 visualizes these classes, and Section 2.2.3 elaborates on different the P2P classes and their resulting topologies.).

CENTRALIZED Connecting the subjects in a centralized topology, the central entity (the service provider) establishes the communication by relaying the messages from sender s_i to recipient(s) r_j. In such a topology, the service provider can infer senders and recipients easily by assessing its routing table.

HYBRID Distributing the functionality of the central entity to a (limited) number of subjects results in a decentralized or hybrid topology. Here, the selected subjects (super-peers) are maintaining a subset of subjects that are only connected to one active subject; the selected subjects are connected among themselves to establish system-wide routing and communication. In a decentralized topology, there is no single subject being able to discriminate all possible senders and recipients.

DECENTRALIZED A full distribution of the functionality of the central entity to all subjects forms a pure P2P system, forming a distributed topology. In such a distributed topology, all subjects may cooperate to establish the communication. This distribution of responsibility requires more effort in the establishment of communication but yields better anonymity protection as the effort for an adversary to collect evidence to identify senders and recipients increases.

> **Summary considering efficiency and anonymity:**
> The topology influences efficiency by facilitating different levels of knowledge to deliver messages. In a centralized topology, a single entity is aware of all subjects and may deliver messages with the best efficiency. In a hybrid topology, a selection of subjects has to cooperate to deliver messages; in decentralized topology, all subjects have to cooperate. Reduced knowledge accompanies the necessity of cooperation in hybrid and decentralized topologies. This reduced knowledge may decrease efficiency as non-optimal decisions may be made.
> Here, anonymity is inversely proportional to efficiency. The fewer knowledge is shared, the better the anonymity as an adversary can collect less evidence to identify senders and recipients.

3.2.4 *Provided Anonymity*

As described in Sections 2.1.2 and 2.1.3, anonymity may be protected in different peculiarities.

A system can be characterized by its strength to protect the anonymity of subjects from protecting only contents (*confidential communication* without anonymity) to hiding the all information about the communication (*untraceable communication* with maximum anonymity). All systems considered as state-of-the-art in this thesis are providing at least anonymity according to level-3 and are, thus, distinguished by the stage of protection from unlinkability to untraceability (see Section 2.1.2 for more details):

- *unlinkability*: senders and recipients may be identified but not both for the same communication message
- *k-anonymity*: senders and recipients are hidden in an anonymity set of at least k other subjects

- *absolute anonymity*: senders and recipients are hidden in an anonymity set that contains all subjects of the system
- *untraceability*: senders and recipients are not distinguishable from other subjects, there is no evidence of communication

> **Summary considering efficiency and anonymity:**
> Anonymity is increased with protection measures like adding cover traffic. Depending on the stage of protection, subjects have to employ different mechanisms. Usually, a higher stage introduces more load to the system and, thus, reduces the efficiency.
> The higher the stage of protection, the better the anonymity.

3.3 ANONYMOUS COMMUNICATION SYSTEMS IMPLEMENTATIONS

Using the anonymization primitives of Section2.4.2 and characteristics of Section 3.2, anonymous communication systems can be described and grouped according to five classes of systems that achieve anonymity following the same idea.

First, the proxy-based anonymization systems are discussed. Then, the more sophisticated systems based on mix networks and OR are discussed. After that, the section introduces the systems that are based on DCnets and randomized forwarding.

The results of the discussions are summarized by Table 2 at the end of this chapter.

3.3.1 *Proxy-based*

The anonymization servcies anon.penet.fi and Anonymizer [4] realize a proxy-based anonymization.

ANON.PENET.FI anon.penet.fi is an anonymous remailer. Subjects send their e-mails to the proxy, where technical information that is possibly identifying the sender is stripped away and replaced with a pseudonym; the mapping of sender-identity and pseudonym is persisted at the proxy.

ANONYMIZER Anonymizer is a similar service, but focussed on anonymous web browsing. Subjects are requesting websites using the proxy, which is relaying the request on behalf of the subject. By that relaying, the recipient does only realize the proxy as initiator of communication but does not learn about the sender's identity (assuming that the content of the message does not leak information).

ASSESSING PROXY-BASED ANONYMIZATION Proxy-based anonymization is comparably efficient and easy to accomplish. However, it lacks effective anonymity protection as it is susceptible adversaries $\mathfrak{A}$ being able to compromise the service provider or to inspect the connections of the network. anon.penet.fi, for example, was shut down after being repeatedly subpoenaed and ordered to reveal sender identities [48]. Proxies establish 1-to-1 communication by relaying communication messages without further intelligence. The connection mode depends on the used implementation and use case, anon.penet.fi relays messages and is, thus, connectionless. In contrast, Anonymizer aims to obfuscate web surfing and relays a connection between sender and recipient. The created topology is centralized with the proxy being the central entity.

3.3.2 *Mix-based*

Mix-based anonymization advanced over many generations that elaborated on different aspects of efficiency, timely behavior (latency), and anonymization.

CYPHERPUNK REMAILERS (TYPE I REMAILER) Cypherpunk remailers [48, 114] are loosely based on Chaum's mix design [23] and advance over the idea of anon.penet.fi by relaying messages over a series of proxies. As cypherpunk remailers do not pick up message padding and shuffling, thus, they are vulnerable to global adversaries.

Cypherpunk remailers introduced the notion of *reply blocks*. These reply blocks enable the recipient to reply to a message without revealing the sender's identity.

MIXMASTER (TYPE II REMAILER) Mixmaster [107, 114] improves the cypherpunk remailers by adding message padding as well as batching and shuffling of messages. Moreover, to also defeat replay attacks, Mixmaster mix nodes check message ids and discard duplicate messages.

Unlike the cypherpunk remailers, Mixmaster does not include return blocks excluding the ability for anonymous replies.

MIXMINION (TYPE III REMAILER) Mixminion [34] is the latest improvement on the remailer type of anonymous communication. Mixminion improves the efficiency of the replay-prevention by introducing periodic key rotation; thus, mix nodes have to memorize message-ids only while the respective key is valid. Also, the bootstrapping is simplified by providing available mix nodes using public *directories*.

Mixminion re-introduces the reply blocks with one-time usage restriction to improve the anonymity of anonymous replies.

WEB MIXES (JONDO, AN.ON, JAP) Web Mixes [15], also known as JonDo, AN.ON, and Java Anon Proxy (JAP) is both a commercial and free implementation of mix-based anonymization with cover traffic. Web Mixes is based on the type I–III remailers; Web Mixes improve the anonymization by adding cover traffic and also enable synchronous communication like web surfing. Subjects have to send and to receive a message in every round. If one of these messages is missing, cover traffic is generated by subjects (in case of a missing message towards the mix chain) or the mix nodes (in case of missing messages toward subjects). To further improve anonymity, Web Mixes also employs fixed message sizes. Thus, smaller messages are padded and larger ones are split.

VUVUZELA Vuvuzela [146] combines a mix network with cover traffic to anonymize communication. Subjects establish communication using a specific establishment protocol, if the recipient accepts, a rendezvous point is assigned to this communication. The rendezvous point is valid for one round only and re-assigned each round to prevent long-term intersection attacks. Subjects have to check for communication-requests every round and are only able to be involved in one communication at a time. Subjects have to generate cover traffic by sending a message to a disposal rendezvous point in each round in which they are not emitting a message.

RIFFLE Riffle [87] combines the message shuffling of mix networks with OR and cover traffic. Mix nodes in Riffle are able—and required to—verify the shuffling of the preceding mix node.

The sender in Riffle is hidden using cover traffic; thus, every subject is required to send a message in every round, if subjects are not involved in communication, they have to generate a dummy message. Recipients are hidden by delivering messages using either broadcasts (if a multitude of subjects are recipients) or private information retrieval (if only a few subjects are recipients). Using latter option, all subjects are required to emit a private information retrieval request to cover actual recipients.

CMIX cMix [24] is a recent approach to enable large-scale anonymous communication using a mix network. In contrast to traditional mix networks, mix nodes are not cascaded but pooled behind a network handler. The network handler relays messages from senders to the mix nodes as well as from the mix nodes to the recipients. The mix nodes perform their shuffling and cryptography based on precomputed key materials and avoid costly public-key cryptography.

cMix is intended to be embedded in PrivaTegrity, a project that tries to balance anonymity and accountability to end the *Crypto War* [64]. In PrivaTegrity, the operators may be convinced about illegal actions of subjects by authorities and can decide to cooperatively deanonymize this subject.

ASSESSING MIX-BASED ANONYMIZATION Mix networks or mix-based anonymization services improve the anonymity protection of proxies by relaying messages over multiple relays. Anonymity is also increased by shuffling the messages at each mix node. However, by this end-to-end design, mix networks mostly do not support native group communication. As all approaches but cMix rely on public-key encryption subjects are not required to establish explicit connections to the mix nodes. All mix network-based approaches rely upon special entities to perform the anonymization. Thus, they establish a decentralized topology. Anonymity at the level of unlinkability is achieved by this class of systems. If additional measures, for example, cover traffic, are taken, untraceability is possible. The protection of anonymity is independent of subject's contribution, mix network-based approaches can provide anonymity in dynamic environments. However, as mix networks blend messages in larger groups of messages to break the coupling, they require a minimum number of senders in each round to achieve reasonable anonymity.

3.3.3 *OR-based*

OR-based anonymization is based on the same idea of multi-layer relaying as mix networks do. The anonymization primitive OR (see Section 2.4.2.3) is most prominently realized by Tor [46].

TOR Tor [46] improved OR and is the first and only widely accepted anonymization service. To improve latency, Tor waives shuffling of messages, padding, and traffic shaping. Subjects establish a usually *circuit* of three Tor-nodes, each of which shares a secret key with the subject. These keys are then used to perform multiple layers of encryption on messages so that each Tor-node can access a single layer of encryption. In principle, Tor is able to be used entirely distributed without any central entity. In reality, Tor is depending on centralized, though replicated, directories that are used to bootstrap subjects and inform them about available Tor nodes.

Tor was initially designed to connect subjects to other external subjects, e.g., web services. *Hidden services* in Tor enable subjects to communicate with other subjects that are located within Tor; subjects providing such a hidden service, for example, a bulletin board, establish a circuit that ends at a Tor node that listens at a specified port. By connecting to this Tor node on the specified port, using an own Tor circuit, subjects are able to use the hidden service with bilateral anonymity.

Over time, Tor has been analyzed and improved with respect to its performance, efficiency, and resistance against adversaries [2, 3, 7, 8, 56, 78, 108]. In general, Tor is vulnerable against global adversaries that are able to inspect the two endpoints of a circuit. The probability of being able to inspect or control these endpoints can be increased by an adversary, as, for example, the routing mechanisms and subjects in Tor prefer high-bandwidth Tor nodes and attacks like the congestion attack [56, 108] shift the selection probabilities of Tor nodes.

MTOR MTor [94] is a extension of the traditional Tor by Dingledine [46] which aims to provide anonymous group communication at large scale. MTor establishes a multicast tree that is composed of Tor-internal subjects; to send a communication message, a subject sends the message towards the root of the multicast tree to distribute the message accordingly to recipients.

Since MTor is fully integrated in the original Tor system, it relies on the same assumptions as Tor does and inherits the weaknesses against global adversaries.

PISCES Pisces [106] leverages the concept of *social trust* between subjects, i.e., leverages out-of-band knowledge and trust between subjects.

Based on social connections between subjects, Pisces selects connections and establishes OR-like circuits for anonymization. Pisces employs a routing scheme that is based on a distributed hash table (DHT) to ensure efficient message dissemination. Subjects regularly check whether neighbors are excluding them from their routing tables, for example, to shift the routing of messages in their favor, and exclude them from their own routing table in a tit-for-tat manner. Moreover, Pisces limits the frequency of changes of the underlying network to prevent adversaries from spawning new identities frequently ("whitewashing").

ASSESSING OR-BASED ANONYMIZATION OR-based anonymization services waive time-consuming protection measures such as batching and shuffling of messages as well as traffic shaping to enable low-latency communication.

While the topology of OR-based anonymization mechanisms can be fully distributed mechanism-wise, Tor is relying on directory servers for bootstrapping and announcing hidden service locations, and is biased towards high-bandwidth Tor nodes. Thus, Tor establishes a decentralized topology.

This avoidance of protection measures in combination with biased routing mechanisms makes Tor vulnerable to global adversaries and various attacks (e.g., [7, 8, 56, 108]).

Due to the circuit design that provides anonymous end-to-end-connectivity, is native group communication not feasible using OR-based anonymization in its current manifestations. However, using extensions like MTor, group communication is realizable.

3.3.4 *DCnet-based*

DCnets provide provable anonymity at the cost of efficiency. Therefore, the two main contributions in the class of DCnet-based anonymization focus on improving the efficiency.

HERBIVORE Herbivore [61] is an approach to increase the efficiency of DCnets by splitting the subjects into smaller interconnected groups, each of which builds a smaller DCnet.

Subjects in Herbivore are joining a group that is randomly selected by a central topology control entity. The groups enable their internal communication using a DCnet and exchange inter-group communication using P2P network that connects the groups. Groups are restricted in size with a parameter k: groups larger then $3k$ are split, groups smaller than k are closed. As groups are limited in size, the communication overhead is also limited.

DISSENT Dissent [30, 149] combines the shuffling of mix networks with DCnet. Dissent employs multiple servers that are responsible for the anonymization.

Groups in Dissent are created by subjects that select a set of servers and define an admission policy. In opposite to traditional DCnet approaches, in Dissent the secrets are not shared amongst subjects but each subject shares secrets with all of the servers that are assigned to the group of the subject. By sharing the secrets with servers, the subjects only exchange messages with the servers of the group which lowers the overhead drastically. This enables Dissent, in contrast to traditional DCnets, to not have all subjects sending a message in every round—servers can exclude keys of not participating subjects easily without rendering the messages of remaining subjects useless.

As Dissent is utilizing dedicated servers for the anonymization, native group communication is not possible.

ASSESSING DCNET-BASED ANONYMIZATION DCnets provide low latency anonymization as, in contrast to mix networks and OR-based anonymization, the anonymization is based on information coding instead of on time-consuming obfuscation. The necessary key-exchange establishes connections between subjects that are—at least to a certain degree—stable and reused for multiple messages. The DCnet design itself allows anonymization in a fully distributed topology. Dissent, however, relies on dedicated servers to improve the efficiency of the DCnet, thus establishing a decentralized topology. By utilizing information coding, DCnet-based approaches are able to provide provable anonymity to the level of untraceability. However, the discussed systems Herbivore and Dissent enable dynamic subject behavior; the achievable anonymity set sizes depend on the long-term availability of subjects. Strong dynamics may degrade anonymity over time when facing long-term communications. DCnets distribute messages among all participants; thus, all subjects in possession of the key material can access all communication messages—native group communication is inherently available. However, as Dissent, for example, is utilizing dedicated servers to improve the efficiency of the anonymization, native group communication is no longer working.

3.3.5 *Further Approaches*

Besides the anonymization systems that are based on the four primitives treated in Sections 3.3.1–3.3.4, there are systems that achieve anonymization based on other approaches like randomized forwarding, cover traffic, and restricted (*friend-to-friend*-like) connections.

P^5 P^5 [133] establishes anonymous communication using the anonymization primitive cover traffic.

Subjects in P^5 are forming groups along a binary tree; recipients are hidden in their respective groups. A sender emits the communication message to the group that shares the longest possible shared prefix with the recipient(s), the addressing scheme follows from the binary tree. Messages are send to the whole group of the respective recipients. To provide anonymity for senders, all subjects generate cover traffic.

Subjects may join multiple groups to improve efficiency; these subjects may provide their lateral connections to other groups for direct message relaying.

RIPOSTE Riposte [29] establishes anonymous communication based on cover traffic and a combination of secure multi-party computation and private information retrieval.

Riposte aims to provide group communication, assuming the existence of few senders and many recipients. To disseminate a message, Senders perform a reverse private information retrieval to submit a message secretly to the servers. To increase the anonymity set for senders, all subjects are required to produce cover traffic in form of an empty reverse private information retrieval request. The servers keep the messages available in a form of a public bulletin board.

Recipient anonymity is not considered in Riposte.

ANONPUBSUB AnonPubSub [37, 38] utilizes the ideas of multi-hop relaying from mix networks and OR. For increased recipient anonymity, the primitive randomized forwarding is included and combined with topology shuffling.

AnonPubSub employs the Pub/Sub communication scheme to support native group communication; anonymity is provided by additional mechanisms to ensure anonymous group establishment. Using *probabilistic forwarding*, Recipients may forward communication messages to subsequent neighbors to cover their own active role in the communication. AnonPubSub enables overlay-position swaps by neighborhood exchanges between subjects with the mechanism *shell game*; hereby, AnonPubSub avoids leakage of role-information from structural properties of the groups

Sender anonymity is not considered in AnonPubSub.

3.4 CONCLUSION

This chapter introduced a set of characteristics that is setting the scene for the core properties efficiency and anonymity.

Group communication raises efficiency as a challenge, especially in anonymous communication systems where information sharing is restricted. Late message duplication reduces the load in the system by pushing the creation of the individual messages as close to the recipients as possible. Without native group communication, the individual messages are created directly at the sender. Similarly, do the connection mode and the topology influence the ability of the communication system to protect the anonymity of the subjects while realizing efficient message dissemination. Lastly, the provided anonymity does not only evaluate the level of anonymity that an anonymous communication system can yield to its subjects but can also be used as an estimator for possible efficiency.

These characteristics at hand, this second part of this chapter discussed state of the art systems in the field of anonymous communication. The discussion is structured along the used core anonymization primitive (see Section 2.4.2) of the respective communication systems.

Table 2 summarizes the comparison of the discussed anonymous communication systems. The key lessons from this chapter are as follows:

- Native group communication is hardly supported. Most systems require the senders to disseminate one-to-one messages repeatedly.
- Enabling dynamic user behavior, i.e., being able to cope with user churn, comes with the dependency on dedicated anonymization subjects. These subjects perform the anonymization on the subject's behalf and are to be stable participants of the communication system.
- Unlinkability is the major anonymity level provided, i.e., the current anonymous communication systems accept that participation in the system is not sensitive and sometimes even identification as a sender (or respectively recipient) is acceptable if both roles are not identifiable at the same time and cannot be linked to the same message.
- Efficiency is hardly considered when developing anonymous communication. Even when considered, the focus is mainly time-related efficiency.

With these learnings, the next chapter provides a novel overlay establishment mechanism that connects senders and recipients while considering efficiency *and* anonymity.

Table 2: Comparison of State-of-the-Art Anonymization Services

Class	System	Group Communication	Server Independent	Dynamic Subjects	Anonymity	Efficiency
Proxy	anon.penet.fi	✗	✗	✓	✗	✢
	Anonymizer	✗	✗	✓	✗	✢
Mix	Cyph. Remailer	✗	✢	✓	✢	✗
	Mixmaster	✗	✢	✓	✢	✗
	Mixminion	✗	✢	✓	✢	✗
	Web Mixes	✗	✢	✓	✢	✗
	Vuvuzela	✗	✢	✓	✢	✗
	Riffle	✗	✢	✓	✢	✢
	cMix	✗	✢	✓	✢	✢
OR	Tor	✗	✢	✓	✢	✢
	MTor	✓	✢	✓	✢	✢
	Pisces	✗	✓	✢	✢	✗
DCnet	Herbivore	✓	✢	✓	✓	✢
	Dissent	✗	✢	✓	✓	✢
Further	P^5	✓	✓	✢	✓	✢
	Riposte	✓	✢	✓	✓$_{\mathcal{S}}$/✗$_{\mathcal{R}}$	✢
	AnonPubSub	✓	✓	✢	✢	✓

✗: 1-to-1; ✓: 1-to-m

✗: centralized; ✢: decentralized; ✓: distributed

✗: no dynamics; ✢: limited dynamics; ✓: dynamic

✗: no protection; ✢: unlinkability; ✓: untraceability

✗: not considered; ✢: limited consideration; ✓: considered

4 EFFICIENT AND ANONYMOUS COMMUNICATION

This chapter introduces a novel overlay establishment mechanism for anonymous group communication that is based on P2P-based Pub/Sub systems. The overlay establishment mechanism is based on ACO and establishes near optimal connections between senders and recipients of information; with the core optimization goal being the communication efficiency. Nonetheless, anonymity is essential and thus, has to be preserved at a reasonable level despite the efficiency improvements.

This chapter is structured as follows: Section 4.1 motivates this chapter and discusses overlay establishment and current approaches for overlay establishment in brief. Section 4.2 introduces the concept of ACO. It is known since the 1990s and has been adapted to various fields of heuristical optimization. Moreover, this section also shows how and why ACO is applicable to anonymous group communication systems if adapted as shown in this chapter. Section 4.3 introduces a novel mechanism for establishing communication overlays by reducing the number of connections between senders and recipients resulting in improved efficiency. This section discusses the algorithm of ACO-based overlay establishment in three phases with the general establishment and as well as the specifics controlling the overlay optimization. Section 4.4 discusses the quantitative simulation results for efficiency, anonymity, and the trade-off between both efficiency and anonymity. Section 4.5 draws conclusions for this chapter and summarizes the findings.

4.1 MOTIVATION

Communication overlays assume the functionality of a service provider. As such, communication overlays connect senders and recipients per communication group and enable message transmission between them. By establishing the routing information, communication overlays outlines the efficiency and anonymity of the communication system.

In anonymous communication, efficiency and the provided anonymity level are two conflicting optimization goals. Efficient communication requires topological information to disseminate message along reasonable communication paths which avoid unnecessary detours. In contrast to efficient communication, anonymous communication requires minimizing the amount of shared information. Adversaries may collect additional evidence from this shared information which enables them to discriminate senders and recipients from other subjects. Also, efficient shortest-path overlays, as they are currently used in [37, 53], leak information about the senders and recipients as the communication paths are deterministic. Deterministic overlays enable adversaries to learn from topology and overlay establishment mechanism and to draw conclusions about potential senders and recipients.

Overlay establishment mechanisms can establish two types of overlays, *persistent* and *transient* ones. Communication systems with persistent overlays reuse them to transmit multiple messages; in contrast, communication systems that use transient overlays establish their communication paths for each message individually. Both types of overlays have advantages and disadvantages. Persistent overlays enable the communication system to incorporate additional anonymization measures; nonetheless, persistent overlays provide an adversary with more possibilities to collect evidence as it uses the same paths multiple times. In contrast, transient overlays impede these efforts of an adversary by reconstructing (possibly) new paths for every message. This reconstruction, however, also weakens the applicability of anonymization measures.

MECHANISMS TO ESTABLISH COMMUNICATION OVERLAYS

Currently, the following approaches are typically used to establish (anonymous) communication overlays in P2P systems [53]: *advertisement*-based and *subscription*-based protocols. Following an advertisement-based approach, participants can *flood* their respective interests (*advertising* their intent to send information) and their counterparts reply (*subscribing* to express their interest to receive information; advertisement-subscription-based overlay establishment). The established overlay equals a shortest path dissemination tree.

Following a subscription-based approach, participants can disseminate their respective interests, for example, using a *random* *walk*, without preceding advertisement (subscription-based overlay establishment).

Both approaches do not focus on efficiency during the establishment of the overlays. Flooding optimizes the overlays regarding delay by incorporating only shortest paths; random walks are used to increase the costs for an adversary by avoiding deterministic path selection but do also increase the length of dissemination paths.

4.2 ACO – A LOOK AT THE PAST AND TO ITS APPLICATION TO ANONYMOUS COMMUNICATION

Dorigo et al. [47] propose ACO as a method for stochastically solving optimization problems. ACO is an analogy to the mechanics of real ant colonies—*ants explore paths independently in their search for food*. However, ants favor joint paths that are indicated by a pheromone trigger. ACO is grounded on distributed computability, and positive feedback, i.e., ACO does *not* need global knowledge to find a solution, and it will discover efficient solutions rapidly [17, 73, 74].

4.2.1 *Real World Analogy – Basic Idea*

ACO is an analogy to the behavior of real-world ants foraging. Ants are social beings, working together to ensure the survival of the whole colony—a goal that the ant society considers more important than the needs of a single ant.

When roaming between their nest and food sources, ants deposit pheromones on their paths to mark them. Ants can smell other ants' pheromones and prefer paths that are used by many ants. As a result of this and the circumstance that shorter paths are likely to be used by more ants in the same time compared to longer paths, ants eventually establish short and efficient paths between food sources and their nest.

When ACO is adapted to decentralized, anonymous communication systems, ants can be seen as, for example, intelligent messages or (mobile) agents.

4.2.2 *Applications of ACO for Heuristical Optimization*

By providing formal requirements for the problem design, setup, and ant properties, Dorigo et al. [47, 96] widened the applicability of ACO and provided a first overview on the field of ACO-based application solving a multitude optimization problems. Blum et al. [17] provide a survey of ACO-based applications and show that ACO is applicable, for example, to problems that can be mapped to the Travelling Salesman Problem and to scheduling problems as well. Moreover, ACO is already used to construct P2P overlays [5] and in telecommunication networks [42]. However, in both applications, ACO is supported by additional knowledge about the network topology, for example, the global topology. Anonymous communication systems aim to avoid this additional information as this kind of topology information may assist an adversary who tries to focus their attack. In file-sharing applications [126, 127], ACO is used to establish a connection between the publisher of a file and the respective requesters. In this scenario, however, the connections are only used in a one-to-one manner; therefore, the concepts lack efficiency when they are applied to anonymous group communication with one-to-many connections.

4.2.3 *Optimal Solutions*

The concept of ACO can compute *optimal* solutions for some domains and with some limitations. Gutjahr [73, 74] derived a graph-based ACO system and showed its convergence to the optimal solution. For that, ants perform a random walk on a construction graph containing all possible solutions. Gutjahr proposes an "elitist" pheromone spreading strategy within which only the best path gains pheromones. Thus, if the optimal path is selected by an ant eventually, this path will succeed over the other solutions in the end, and ACO converges to the optimal solution. Gutjahr poses requirements to this convergence behavior such as having an optimization problem that can be represented by a graph, possesses a unique optimal solution, a single fixed start node for all ants, and requires that all random walks of ants be free of loops and of maximal length. The overall probability to converge to this solution is $1-\epsilon$ for arbitrarily small ϵ by varying the parameters of the ACO-heuristic [73].

Later, Gutjahr was able to show that the convergence of ACO to the optimal solution can be guaranteed [74]. To achieve this guarantee, Gutjahr introduced a cooling mechanism, similar to the metaheuristic *simulated annealing* [83]. With this cooling, the ants are restricted in their random walk, thereby improving their probability to converge to the optimal solution.

Nonetheless, both approaches rely on parameter-tuning that is hardly possible in distributed systems without a global control entity. Moreover, the uniqueness of the *optimal* communication overlay in the overlay establishment can mostly not be guaranteed. The key lesson of this section is that ACO can establish efficient solutions even though there is no guarantee for its convergence to an optimal solution.

4.2.4 *ACO's Adequateness on Anonymous Group Communication*

ACO is a metaheuristic that is used to approximate optimal solutions to hard problems, such as the traveling salesmen problem, graph coloring. Thus, ACO stands in row with other metaheuristics as, for example, simulated annealing [83], particle swarm optimization [82], genetic algorithms, and evolutionary programming, see for example [49].

A communication system establishes communication overlay by spreading topological information about senders and recipients. Anonymous communication systems try to minimize this information and additionally try to avoid that a single subject, for example, the service provider, can gather all information. Despite the minimization of shared information, the established overlay should minimize the number of connections in the overlay to ensure efficient communication. With ACO, each subject in the system handles agents based on their local knowledge, thereby limiting the information spread.

The convergence guarantees of Gutjahr [73, 74] are not applicable as neither the uniqueness of the optimal overlay can be guaranteed nor is the necessary parameter tuning possible without leaking information that would overturn anonymity. Nonetheless, the concept of ACO is promising to be adapted to the overlay establishment in an anonymous group communication system.

4.3 ANT COLONY OPTIMIZATION TO ESTABLISH COMMUNICATION GROUPS

This section describes the novel adaption of ACO to anonymous group communication and specifically to establish overlays for them. First, the ACO-specific model and terminology are introduced, and a high-level concept provides an overview concerning the ACO-based overlay establishment; after that, the three phases of ACO-based overlay establishment are introduced in detail.

The proposed novel overlay establishment mechanism addresses the following challenges:

- Establish communication overlays that ensure connectivity between senders and recipients.
- The established communication overlays should minimize the number of connections—without involving more than the local knowledge of the subjects.
- The overlays should provide a reasonable trade-off between efficiency and anonymity.

4.3.1 *Model and Terminology*

This section introduces the ACO-specific model and terminology and complements the background in Chapter 2. The notations are summarized in Appendix A.

AGENTS The ants of ACO are used to find solution to an optimization problem. In the scope of this thesis, their digital representations are called *agents* $a \in \mathcal{A}$. Hereby multiple instantiations are possible:

- *Mobile Agents* or intelligent messages that carry their (implementation) code with them. Each subject then enables the agents to execute their code by delivering their data as input to the computations.
- *Containers* that carry only the properties of the agents while subjects themselves have the (implementation) code. Agents enable the subjects to perform the necessary computations by providing the necessary data as input.

Both instantiations will produce equivalent results. However, the trust to perform the required computations honestly and provide the correct data is shifted either onto the subjects or the agents.

This thesis uses agents only as containers. Subjects are equipped with the implementation, and perform the necessary computations using the properties carried by the agents.

All subjects organize the agents that they received in two sets that correlate with the phase of the respective agents:

- $\mathcal{A}_{\mathrm{Disc}}$ that holds agents looking for senders (*Phase 1*)
- $\mathcal{A}_{\mathrm{Ret}}$ that holds agents returning from senders (*Phase 2*)

The full set of agents $\mathcal{A}$ is the union of the sets $\mathcal{A}_{\mathrm{Disc}}$ and $\mathcal{A}_{\mathrm{Ret}}$.

SENDERS AND RECIPIENTS AND THEIR ROLES Senders and recipients have different roles depending on the implementation of the ACO-based overlay establishment. Similar to the advertisement-based or subscription-based overlay establishment in Pub/Sub [53], senders or recipients may initiate the overlay establishment using ACO.

The initiator takes the role of an *ant colony* and creates agents that search for *food* which is represented by the respective counterpart(s) of the initiator. In this thesis, recipients take the role of ant colonies and initiate the overlay establishment; senders represent food and are therefore the destination of created agents.

The usage of such a recipient-initiated overlay establishment reveals advantages in this scenario:

- Multiple recipients *collaborate* to explore the network quickly, successful recipients speed up the search of other recipients by sharing the pheromones
- Recipients may establish their connection to the communication overlay by *connecting to an already connected recipient*.

If an application scenario is more suitable for the sender-initiated overlay establishment, senders and recipients have to reverse roles in the ACO-based overlay establishment: Senders emit agents and recipients are the destination of the agents. The overlay establishment mechanism itself does not require additional changes.

PHEROMONES Pheromones τ are deposited by agents in phase 2 and can be used to assess the popularity of a connection; pheromones are directed to indicate the direction of movement and to avoid attract agents towards dead ends. As such, pheromones τ can be expressed towards a neighboring subject $n \in N^-$ by τ_n or on a connection e by τ_e.

Agents in phase 1 bias their exploration of the system based on their preference for pheromones (see the *strictness* parameter and its influence on the *random walk procedure* in the description of phase 1 in Section 4.3.3.)

4.3.2 *Concept*

Adapting ACO to the subscription-based overlay establishment enables reasonably fast overlay establishment: all recipients emit agents that perform their random walk to search for the senders in parallel; these agents help each other to establish the overlay. The agents influence each other by spreading pheromones which bias the random walk of the agents. In contrast to this, in an advertisement-based overlay establishment, only the senders would emit agents. As a result of this, the number of agents sinks and therefore the time until the overlay is established will increase.

The core idea of adapting ACO to the overlay establishment is sharing pheromones of the agents within a communication group; other recipients can connect themselves faster to the overlay by exploiting already established paths and pheromone trails. Moreover, by sharing the pheromones, agents foster path consolidation and reduce thereby the overall message load in the communication system. Figure 22 visualizes the difference between a state of the art communication overlay based on flooding and its ACO-based counterpart. The ACO-based communication overlay uses a lower number of connections and, thereby improves the efficiency of the communication.

The overlay establishment is performed in three stages: the first two by usual *agents*, the third by *activation agents*.

1. *Sender Discovery*: Agents are emitted by recipients to *search and discover* senders. Agents perform a random walk with a (controllable) bias towards existing pheromone trails until they either discover a sender or their TTL expires and terminates the random walk.

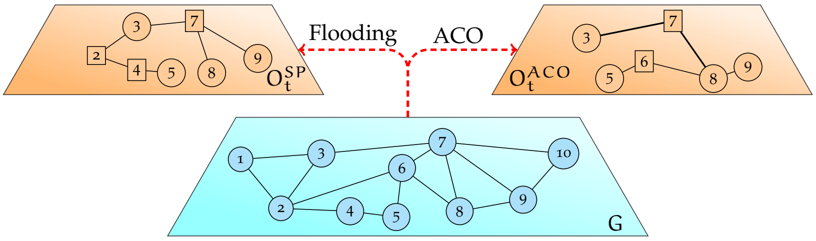


Figure 22: By consolidating the dissemination paths, ACO (right overlay O_t^{ACO}) is able to improve the efficiency compared to the flooding-based overlay (left overlay O_t^{SP}) that relies on shortest paths.

2. *Return to Recipient*: When agents found a sender, they return to their recipient (i.e., they return to their source) on the previously discovered path. On the return path, agents deposit pheromones on connections to bias the random walk of following agents. Over time, participants in the system reduce the number of pheromones on incident connections with a linear factor to avoid convergence to non-optimal overlays.

3. *Overlay Activation*: To ensure stable message dissemination and to terminate the overlay establishment itself, the strongest pheromones trails are "activated" eventually. For that, recipients emit activations agents that strictly follow the strongest pheromone trails towards the senders and activate this path upon their return from the sender to their recipient. With that design, message loss is avoided. Messages from the sender either follow old paths if they are emitted before the potentially new path is activated, or they follow the new paths if they are emitted after the new path is activated.

The following sections introduce the three phases in detail and discuss the configuration parameters and design decisions.

4.3.3 *Phase 1 - Sender Discovery*

In this first phase, each recipient creates agents that explore the system to find senders. These agents are emitted into the communication system and randomly relayed between the participants. Subjects collect the agents that look for senders in the set $\mathcal{A}_{\mathrm{Disc}}$. Subjects forward agents from the set $\mathcal{A}_{\mathrm{Disc}}$ to random neighbors, i.e., the agents perform a random walk, to discover the vicinity. As a result of this, agents may walk indefinitely without discovering a sender. Therefore, agents maintain a TTL that is decremented after each hop and the walk terminates if the TTL reaches zero.

AGENT CREATION In this first phase, recipients create agents to explore their vicinity to discover senders.

The strictness σ of an agent controls the controls the explorative behavior of agents by weighting the influence of pheromones τ on the transition probabilities of their random walk—which resembles the positive feedback mechanism of ACO. When recipients create new agents, they initialize strictness σ of the agents with a random value to ensure a mixture of exploring and exploiting agents; σ is restricted to $[0.0, 1.0]$.

Larger values of σ reduce the influence of pheromones on the weighted random walk, i.e., larger values of σ foster exploration. In contrast to that, lower values of σ increase the influence of pheromones on the transition probability, reducing the ability of an agent to explore their vicinity, i.e., lower values of σ foster the exploitation of already established pheromone trails.

Equation (8) shows how an agent a weighs the pheromones to calculate the transition probabilities to a neighbor v_k depending on their strictness σ and the pheromone values τ towards the neighbors. Larger values of σ increase the probability of selecting neighbors with higher pheromone values towards them. Figure 23 visualizes the probability of choosing a specific neighbors depending on an agent's strictness σ and the neighbor's pheromone value τ. In this example, four neighbors (a–d) have pheromones ($\tau_a = 5, \tau_b = 15, \tau_c = 30, \tau_d = 50$) that sum up to a value of 100. The probability of choosing a specific neighbor is then computed using Equation (8).

$$p_{\tau,\sigma} = \frac{\tau_{v_k}^{a.\sigma}}{\sum_{v_m \in N^{-}} (\tau_{v_m}^{a.\sigma})} \tag{8}$$

Proposition 1 *Drawing a random strictness value σ from [0.0, 1.0] ensures a mixture of exploring agents (smaller values of σ) and exploiting agents (larger values of σ).*

The TTL of agents is initialized with a value that estimates the apl (see Section 2.2.1.3) of the logical underlay G, i.e., initialized with an estimate of the average shortest path in the system. This enables agents to connect recipients without risking

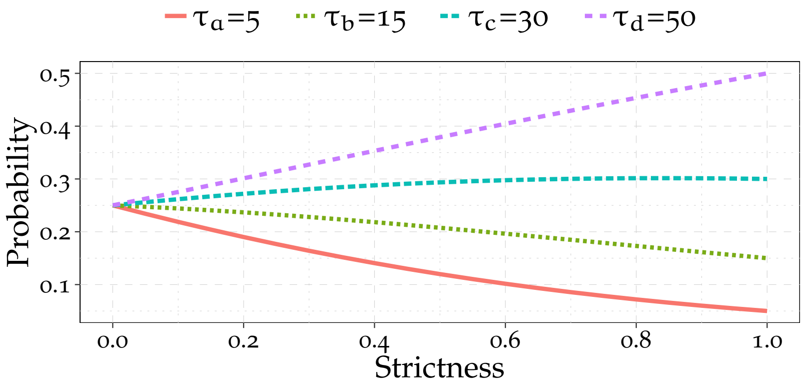


Figure 23: Probability of selecting a neighbor depending of the strictness σ of an agent a and the pheromones τ towards the neighbor

to congest the communication system with agents that do long random walks Nonetheless, limiting the agents' TTL may render some sender-recipient combinations not to be reachable by agents. For that, recipients increase the TTL-value of newly created agents over time. Limiting the TTL with the diameter of the logical underlay G is also not sufficient: the agent's random walk to discover a sender is not restricted by the diameter—agents can choose longer paths to utilize shared path segments over the shortest possible connection. Nonetheless, an upper limit of the TTL is required to avoid too lengthy paths. An upper limit of, for example, three to four times the diameter seems reasonable—however, this upper limit needs to be adapted to the specific application scenario. For both the apl and the diameter rough estimates are sufficient; these estimations can be derived from the type of the logical underlay and its size (see Section 2.2.4 and [1, 90]).

The TTL of new agents is therefore given by Equation (9) as the maximum of the apl, i.e., the lower limit of the TTL, and the minimum of the current round and the upper limit. The upper limit is configured to be a multiple of the diameter of the logical underlay. The round expresses the time that the recipient is already looking for senders. The recipient may translate the round rd into hops by multiplying it with the average connection latency (see Section 2.2.2).

$$ttl = max(apl, min(rd, ul)) \tag{9}$$

Table 3 reports exemplary apls and diameters of social networks, random networks, and RGGs and as well as the resulting limits of the TTL. These values show that the TTL of early agents is configured to be comparably low which avoids the heavy load of long random walks in the beginning; recipients, however, enable later agents to explore the system more extensive by raising their TTL. In later phases, the longer random walks are not problematic as agents are then biased towards existing pheromone trails.

Proposition 2 *An* increasing *TTL value ensures that the communication system is not overloaded with agents in the beginning but also ensures that all recipients reach the senders eventually when the TTL has reached the necessary value to enable agents to travel the distance.*

Proposition 3 *The apl as initial TTL configuration enables agents to reach senders on average shortest paths. Raising the TTL over time to four times the diameter will enable agents to reach all senders eventually to establish a connected communication overlay.*

Table 3: Apl and diameter of social networks, random networks, and RGGs and the resulting minimal and maximal TTL of agents

Logical Underlay G	apl(G)	diam(G)	min(TTL)	max(TTL)
social	≈3.649	≈6.15	4	24
random	≈5.515	≈12.779	6	48
RGG	≈6.248	≈15.755	7	60

random walk procedure Procedure 1 summarizes the random walk that is performed by each agent a_j at their current position v_i, i.e., Procedure 1 reflects the decisions made by the subject v_i that is currently homing agent a_j.

Before relaying the agent to a subsequent neighbor, the subject v_i checks in lines 2–3 of Procedure 1 if they are a sender with respect to the agent's topic $t \in T$ of interest. If v is a sender (that is $v_i \in \mathcal{S}_t$), v_i adds the agent a to the set of returning agents $\mathcal{A}_{Ret}$ to initiate the second phase of the agent.

After checking whether the current subject is a sender or not, lines 4–5 of Procedure 1 evaluate the agent's TTL and whether the agent established a loop or not. When the TTL hits 0, the agent is removed. Also, subject v_i assesses whether agent a establishes a loop or not. When a loop is established, the agent

Procedure 1: walkAgent($\mathcal{A}_{disc}$)

```
   /* $v_i$: subject walkAgent($\mathcal{A}_{Disc}$) is called on        */
   Input: A finite set $\mathcal{A}_{Disc} = \{a_1, a_2, \dots, a_n\}$ of agents looking
          for senders
1  for $a \in \mathcal{A}_{Disc}$ do
2      if $v_i \in \mathcal{S}_t$ then
3          $\mathcal{A}_{Ret} \Leftarrow a$
4      else if $ttl = 0 \mid\mid a_j.loopDetetected()$ then
5          $removeFromSystem(a)$
6      else
7          $v_k \Leftarrow selectNextHop(v_i)$
8          $a.decrementTTL()$
9          $a.moveTo(v_k)$
10 return
```

is removed as well. This assessment can be performed by evaluating whether subject v has already forwarded the agent a, i.e., checking whether the unique agent-id of a is already known to v_i. Another possibility is the usage of anonymity-preserving hash-chains [38], with this a random value is included in each agent. Every subject hashes this value upon relaying the agent and stores the resulting hash value also locally. By checking the hash value of an agent with the stored hashes (and the results of the next n applications of the hash functions on the stored values) loops can be detected.

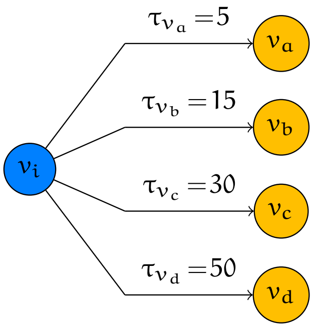
After that, in line 7 of Procedure 1, the agent's next hop is selected from the neighbors of the current subject v_i. Equation (10) formalizes the transition probability towards a specific neighbor v_k which is computed for all neighbors of v_i. This probability is composed of the standard random walk probability and the probability $p_{\tau,\sigma}$, i.e., the weighted influence of pheromones τ from earlier agents.

$$p_{v_k} = 0.5 \cdot \left(\frac{1}{|N^-(v_i)|} + \frac{\tau_{v_k}^{a_j.\sigma}}{\sum_{n_m \in N^-(v_i)} (\tau_{n_m}^{a_j.\sigma})} \right) \tag{10}$$

The standard random walk probability ($1/|N^-(v_i)|$) ensures that each connected neighbor may be selected and given equal importance in the absence of pheromones. Thus, each neighbor has the base probability $1/|N^-(v_i)|$ of being selected as the next hop of an agent—with $N^{-}(v_i)$ being the set of neighbors of v_i (see Section 2.2.1.3).

The pheromone influence is then added to the probability using the probability $p_{\tau,\sigma}$ which is introduced in Equation (8).

Both probabilities are equally weighted in Equation (10) and normalized to achieve a probability value in $[0.0, 1.0]$.



$$p_{v_a}=0.5\cdot\left(0.25+\frac{5^{0.8}}{50.4112}\right)\approx 0.1609$$

$$p_{v_b}=0.5\cdot\left(0.25+\frac{15^{0.8}}{50.4112}\right)\approx 0.2116$$

$$p_{v_c}=0.5\cdot\left(0.25+\frac{30^{0.8}}{50.4112}\right)\approx 0.2757$$

$$p_{v_d}=0.5\cdot\left(0.25+\frac{50^{0.8}}{50.4112}\right)\approx 0.3518$$

$$\sum_{n_m\in\{v_a,v_b,v_c,v_d\}}(\tau_{n_m}^{0.8})\approx 50.4112$$

Figure 24: Transition probabilities to the neighbors of v_i. Agent a_j has a strictness value of $\sigma = 0.8$ and the neighbors a–d have pheromone values $\tau = \{5_a, 15_b, 30_c, 50_d\}$.

Example 20: Figure 24 shows the neighbor selection according to Equation (10) for subject v_i and four neighbors a–d. The pheromone amounts on the connections are $\tau = \{5_a, 15_b, 30_c, 50_d\}$, the agent's strictness is $\sigma = 0.8$. The resulting transition probabilities are $p = \{0.1609_a, 0.2116_b, 0.2757_c, 0.3518_d\}$. The next hop is drawn according to these probabilities.

Varying the agent, and therefore varying the strictness value σ, shows the expected changing probabilities:

σ	p_{v_a}	p_{v_b}	p_{v_c}	p_{v_d}
0	0.2500	0.2500	0.2500	0.2500
0.25	0.2133	0.2413	0.2633	0.2821
0.5	0.1849	0.2288	0.2718	0.3145
0.75	0.1643	0.2145	0.2755	0.3458
0.8	0.1609	0.2116	0.2757	0.3518
1	0.1500	0.2000	0.2750	0.3750

4.3.4 *Phase 2 - Return to Recipient*

When an agent a arrives at a sender $s \in \mathcal{S}$, s initiates the return phase of the agent a. In this phase, the agent will return to its recipient, i.e., the agent's source subject. This section discusses the walk of agents to their recipients using Procedure 2 which describes the steps of this second phase.

RETURN PATH & PHEROMONE ASSIGNMENT In phase 1, agents are recording their paths. In the second phase, they return to their recipient by reversing the saved paths which is described in Lines 1–8 of Procedure 2. Subjects are collecting the returning agents in the set $\mathcal{A}_{Ret}$.

```
Procedure 2: returnAgent($\mathcal{A}_{Ret}$)
   /* $v_i$: subject returnAgent($\mathcal{A}_{Ret}$) is called on      */
   /* $v_{i-1}$: subject from which the agent came to $v_i$            */
   Input: A finite set $\mathcal{A}_{Ret} = \{a_1, a_2, \ldots, a_n\}$ of agents
          returning to their recipients (sources)
1  for $a \in \mathcal{A}_{Ret}$ do
2      $v_{i+1} \Leftarrow a.getNextHopHomewards()$
                    // Pheromone Assignment
3      $v_i.\tau[v_{i-1}] \Leftarrow v_i.\tau[v_{i-1}] + a.\tau$
4      for $v_j \in N^-$ do
               // Diffuse pheromones to neighbors
5          if $v_j \neq v_{i+1}$ then
6              $v_j.increasePheromoneTo(v_i, a.\tau \cdot d_{1h})$
7      $a.moveTo(v_{i+1})$
                    // Evaporate pheromones
8  for $v_j \in N^-(v_i)$ do
9      $v_i.\tau[v_j] \Leftarrow \gamma \cdot v_i.\tau[v_j]$
10 return
```

While returning to their recipient, agents deposit pheromones to inform later agents about the discovered path. The pheromones are directed and placed towards the sender, i.e., when handling a returning agent, subjects place pheromones on the connection towards the preceding subject on the path of the agent. As in the basic ACO, the optimization of communication structures is based on this feedback loop. Statistically seen, more agents will travel on shorter paths and, thus, will deposit more pheromones on these paths. This higher pheromone count increases the probability of selecting this connection for subsequent agents—depending on the agents' strictness σ—and establishes a self-improving loop. Agents deposit pheromones on connections according to Equation (11) and Line 3 of Procedure 2. Each agent places its amount of pheromones $(a.\tau)$.

$$\tau_{e_m} = \tau_{e_m} + a.\tau \tag{11}$$

DIFFUSION In the original ACO, pheromones diffuse in the vicinity of the connections to enable agents to recognize nearby paths. An adaptation of this diffusion to a decentralized system will introduce a continuous wave of diffused pheromone distribution messages. These messages will massively degrade the efficiency of the system.

Diffusion is restricted to the incident connections of the subject v_i, i.e., pheromones spread only 1-hop, to avoid unnecessary message load. Thus, when a subject forwarded all returning agents $a \in \mathcal{A}_{Ret}$, they will share their pheromone gains with their direct neighbors $v_j \in N^{-}(v_i)$ which add a "splash" of pheromones on the connection towards the subject. Exempt from pheromone diffusion is the actual path of the agent, i.e., the connection from which the agent a arrived at v_i does not receive pheromones through the diffusion mechanism; the same holds for the connection that is used to forward the agent a towards its recipient. The neighbors $v_j \in N^{-}(v_i)$ that receive pheromones through this diffusion mechanism are not propagating the pheromones to their neighbors.

The parameter d_{1h} controls the amount of pheromones that are diffusing as provided by Equation (12). The pheromones τ_{diff} are placed at the incident connections, except those connections that are used by agents in this round.

$$\tau_{diff} = d_{1h} * \tau_{gain} \tag{12}$$

Example 21: Figure 25 visualizes the 1-hop diffusion for a subject v_i. v_i gained 20 additional pheromones (on the connection from v_j) from agents that are forwarded towards v_l and v_k. The remaining neighbors of v_i, i.e., v_x, v_y, and v_z, receive a share of these pheromones. This share is computed by taking the gained pheromones τ_{gain} and scale it with the diffusion factor $d_{1h} = 0.1$. Therefore, the orange neighbors v_x, v_y, and v_z receive pheromones $\tau_{\{x,y,z\}} = 2$ on their connection towards v_i.
The neighbors of v_x, v_y, and v_z (indicated by dashed connections and dashed circles) will not receive a share of pheromones.

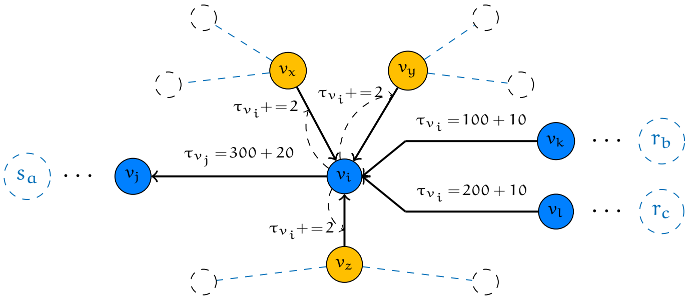


Figure 25: 1-hop Diffusion of Pheromones: The pheromone gain τ_{gain} of v_i from subject v_j is 20. The diffusion to v_i's neighbors v_x, v_y, and v_z is 2 with a diffusion factor of $d_{1h} = 0.1$.
The dashed 2-hop neighbors will not receive pheromones through the diffusion mechanism.

Proposition 4 *The optimization of ACO stays functional even with the limited diffusion to the 1-hop neighborhood.*

EVAPORATION The optimization of ACO requires the system to reduce pheromones over time. Otherwise, the pheromone influence on established paths will influence optimizations by preferring these established paths over more recent ones. Moreover, as denoted in Chapter 5, subjects can be dynamic; thus, subjects can leave an overlay or the system at any point in time and pheromones that indicate a potential path using such a subject should be removed over time. As evaporation does not remove pheromones completely, the remaining pheromones can be used by (later) joining subjects to improve their start-up time, i.e., to connect to existing communication overlays faster.

To this end, the ACO-based overlay establishment mechanism adopts the common ACO part concept and simulates the evaporation to reduce the pheromone amount on the system's connections with a proportional function.

Proposition 5 *The usage of a proportional function ensures that the simulated evaporation does not change the probabilities that are induced by the pheromones. The reduced amount of pheromones, however, enables subsequent agents to find new paths without being influenced by older pheromones.*

Equation (13) formalizes the reduction of pheromones by evaporation. The evaporation factor γ is limited to $[0.0, 1.0]$ and controls the reduction of pheromones. Lower values of γ will foster exploration as the evaporation will lower pheromones significantly; larger values of γ are beneficial for exploitation as the evaporation will reduce pheromones only marginally.

$$\forall e \in \mathcal{E} : e.\tau_{rd+1} = \gamma \cdot (e.\tau_{rd}) \tag{13}$$

The goodness of discovered paths is not affected by this evaporation as all connections with pheromones are reducing the pheromone count by the same factor.

4.3.5 *Phase 3 - Overlay Activation*

In Phases 1 and 2, ACO spreads pheromones in the anonymous communication system. Over time, the pheromones single out the dissemination tree that connects the recipients and senders. To further improve efficiency, each recipient emits *activation agents* that follow the strongest pheromone trail towards the senders and activates this path. This mechanism effectively establishes the communication overlay without lasting dependency on pheromones. Disruptions by leaving subjects due to subject churn are taken care of in Chapter 5.

PERSISTING PATHS TO IMPROVE EFFICIENCY Explicit overlay activation allows to terminate the overlay establishment process as routing tables, and, thus, the communication overlay, are persisted at the subjects.

Paths are persisted by sending activation agents that strictly follow the highest pheromone counts towards senders—the random walk property is disabled for these agents. Just like the usual agents, they memorize the used path. When arriving at a sender, path activation agents return on the same path and place routing details, for instance, predecessor and successor with a group-identifier, at the subjects. Communication messages (m_t^{com}) are then only disseminated using these persisted paths. The information leakage due to the path memorization and explicit overlay activation is discussed in the second part of Chapter 5

Paths are persisted and then updated to account for ongoing optimizations of the ACO mechanism without risking to delay communication without purpose. The duration of the overlay establishment, i.e., the length of Phase 1–3 is configurable.

4.4 EVALUATION

This section analyzes the ACO-based overlay establishment in detail with a simulation study. As this thesis focuses on efficiency concerning the message load, the simulation abstracts from the network layer and focuses on the application layer.

The anonymous group communication system is based on the model in Section 2.2, and is implemented using the Java-based GraphStream [117] graph library. GraphStream models and analyzes static and dynamic graphs. This implementation enables a quantitative simulation providing insights into the efficiency and anonymity using the ACO-based overlay establishment mechanism in comparison to the state of the art flooding-based approach. Also, this implementation is used to analyze the parametrization of ACO. The whole simulation is performed on various topologies of the logical underlay G to show that the results are well-founded and independent from the actual application scenario.

This section is structured as follows: first, the simulation setup and parametrization is given. Second, the evaluation metrics with respect to the ACO-based overlay establishment are discussed. After that, the network configuration is discussed and backed using a simulation study. Then, this section evaluates the parameters of the ACO-based overlay establishment and their individual influence on the communication overlays. Then, this section discusses the resulting communication overlays under the aspects of efficiency and anonymity.

4.4.1 *Simulation Setup*

The communication system uses a logical underlay G that is based on the topologies discussed in Section 2.2.4:

- *Social networks* based on the Barabàsi-Albert model [13]. A configuration with $m_0 = 2$ and $m = 3$ is used to equip the subjects with a social network-like connection structure (see Section 2.2.4.1 for additional details). These social networks occur whenever a social network, e.g., Facebook or Twitter, is used to bootstrap the connections of subjects.
- *Random networks* based on the Erdős-Rényi model [51]. A configuration with an average subject degree of 4 (*random*$_4$) and an average subject degree of 10 (*random*$_{10}$) is used (see Section 2.2.4.2 for additional details). These random networks occur in gossiping-based networks that are maintained, for example, by SCAMP [59].
- *Random Geometric Graphs (RGG)* with dimensions $d = 2$ and a distance threshold $t = 0.1$ are used (see Section 2.2.4.3 for additional details). These networks occur when any distance based networks like WSNs are in focus.

The simulation uses these synthetically generated networks to avoid the influence of special topological situations that may occur in exceptional cases and neighborhoods. The synthetic networks enable repeated—and, thus, statistically sound—simulations without relying on a limited number of real-world networks that are published for research purposes. The single requirement that has to be fulfilled by the networks is connectivity between any pair of subjects, i.e., there is only one connected component that comprises all subjects $v \in \mathcal{V}$.

For the overlay establishment, one sender is randomly picked from the set of subjects $\mathcal{V}$ to form the sender set $\mathcal{S}_t$ and between 15 and 30 recipients are randomly picked from $\mathcal{V}$ to form the recipient set $\mathcal{R}_t$. The subjects in $\mathcal{S}_t$ and $\mathcal{R}_t$ are then connected using the overlay establishment mechanisms; the system executes the overlay establishment mechanism for 500 rounds, if not stated differently.

In these simulations, the communication overlays O_t are created using the following three overlay establishment mechanisms to compare the properties of the resulting overlays. This thesis contributes the ACO-based overlay establishment mechanisms, the flooding-based overlays (described, for example, by Daubert et al. [37]) and the Steiner tree approximation StA (see Section 2.2.5) are used as baselines.

- *ACO:* Establishing communication overlays $\mathcal{O}_t$ using the ACO-based overlay establishment mechanism proposed in this thesis, recipients $r \in \mathcal{R}_t$ are actively searching for senders $s \in \mathcal{S}_t$. By using the pheromone mechanism of ACO, paths are aggregated to reduce the weight of $\mathcal{E}_t$; thus, ACO is reducing the communication overhead that is caused by superfluous communication messages.

- *Flooding:* Establishing communication overlays $\mathcal{O}_t$ using flooding, senders $s \in \mathcal{S}_t$ emit an advertisement that is disseminated to all other subjects in the logical underlay G; upon receiving such an advertisement, recipients $r \in \mathcal{R}_t$ reply with a subscription message. This subscription is sent towards the sender using the shortest path[1]. Flooding-based overlays are the current state of the art to establish communication overlays in P2P-based Pub/Sub systems [37]

- *Steiner tree (StA):* A Steiner tree connects a finite set of *terminals*, the senders $s \in \mathcal{S}_t$ and recipients $r \in \mathcal{R}_t$, using additional *Steiner points*, these additional subjects form $\mathcal{B}_t$. The goal of the Steiner tree establishment is to derive the Steiner tree such that the weight of $\mathcal{E}_t$ is minimized, for instance, by minimizing the number of connections in $\mathcal{E}_t$ (see Section 2.2.5 for more details on the Steiner tree and the used heuristic of Kou et al. [84]).

For each network configuration, 10 network instances are generated. On each network instance, 5 sets of senders and recipients are selected. The simulation is executed on each of the instances; thus, the simulation is repeated 50 times for each type of network.

The adversary is assumed to be in the role of the service provider, i.e., the adversary can observe the complete message flow. The adversary $\mathfrak{A}$ is therefore instantiated to be $\mathfrak{A}_{p,g}$.

[1] The shortest paths refers to the quickest connection, for instance, the one fastest path or the path with least hops. This thesis refers to shortest paths as the path with the least number of hops.

The complete simulation setup is summarized in Table 4. Section 4.4.3 reasons about the chosen network configuration by comparing the properties of networks that are generated with different configurations and network sizes.

Table 4: Simulation Parameters

Parameter	Value
Runs	50
Rounds per run	500
Network size ($\lvert\mathcal{V}\rvert$)	2000
Recipients ($\lvert\mathcal{R}_t\rvert$)	Gaussian distr. ($\mu = 30, sd = 2$)
Adversary ($\mathfrak{A}$)	$\mathfrak{A}_{p,g}$
Network types	{social; random; euclidean}
Connectivity social network	$m_0 = 2$, $m = 3$
Connectivity random network	$\overline{d(v)} = \{4, 10\}$
Connectivity threshold RGG	0.1
Dimensions RGG	2

4.4.2 *Evaluation Metrics*

The simulation is evaluated using the following metrics; these metrics are grouped according to their respective goal: *Efficiency* and *Anonymity*.

EFFICIENCY Regarding efficiency, the evaluation analyzes first whether the overlay establishment with the ACO-based mechanism is successful. Then, the evaluation analyses to which extend the message load can be reduced. Additionally, the evaluation discusses the influence on the communication delay by analyzing the distance between sender and recipients.

- *Receiver Success Ratio* measures whether all recipients $r \in \mathcal{R}_t$ are able to connect to the senders $s \in \mathcal{S}_t$. The success ratio in Equation (14) formulates the receiver success ratio. The receiver success ratio $sr_{\mathcal{R}_t}(O_t)$ is measured by dividing the number of receivers $r \in \mathcal{R}_t$ for which there exists at least one connection $e \in \mathcal{E}_t$, which ends at r_i, by the cardinality of $\mathcal{R}_t$.

$$sr_{\mathcal{R}_t}(O_t) = \frac{|\{r \in \mathcal{R}_t : \exists e \in \mathcal{E}_t : e = (\_, r)\}|}{|\mathcal{R}_t|} \tag{14}$$

- *Weight*, as measured by $w_{\mathcal{E}}(O_t)$ expresses the communication costs for a single message m_{com} from a system-wide perspective. Within this simulation, all connections have a cost of 1, enabling the evaluation to measure the communication costs in the number of handled messages. Generalized, each connection has a weight associated, which can be evaluated using the weight function $w_e(e_i)$—within this thesis, $w_e(e_i)$ returns 1 for all connections. Expanded to the whole overlay, the weight function $w_{\mathcal{E}}(O_t)$ evaluates the costs of the communication overlay—within this thesis, the weight of the communication overlay translates to the number of connections in $\mathcal{E}_t$.

  The lower the weight $w_{\mathcal{E}}(O_t)$, the better the efficiency as the overall messaging load in the system is lower.

$$w_{\mathcal{E}}(O_t) = \sum_{e_i \in \mathcal{E}_t} w_e(e_i) = |\mathcal{E}_t| \tag{15}$$

- *Communication Delay*, as measured by $cd(O_t)$ evaluates the average latency between a sender $s \in \mathcal{S}_t$ and the recipients $r \in \mathcal{R}_t$ when sending a message m_t^{com}. Equation (16) computes the communication average delay between all sender and recipient pairs. For that, the round differences between receiving a message ($r.rcvdAtRd(m_t^{com})$) and sending a message ($s.sendAtRd(m_t^{com})$) are summed and then normalized.

$$\begin{aligned} cd_{avg}(O_t) = \frac{1}{|\mathcal{R}_t|} * \sum_{r \in \mathcal{R}_t} & (r.rcvdAtRd(m_t^{com}) \\ & - s.sentAtRd(m_t^{com})) \end{aligned} \tag{16}$$

  The communication delay can also be abstracted and assessed through deriving the distance between sender and recipients, i.e., computing the path length between the sender and the respective recipients of a communication message. Equation 17 formalizes this approach to measuring the average communication delay.

$$cd_{avg}(O_t) = \frac{1}{|\mathcal{R}_t|} * \sum_{r \in \mathcal{R}_t} p_t(s,r) \tag{17}$$

ANONYMITY

- *Leaf Subjects* are directly identifiable to be interested in a communication topic, i.e., leaf subjects are exposed to an adversary as senders $s \in \mathcal{S}_t$ or recipients $r \in mathcalR_t$. For that, an adversary needs learn the topology of O_t. A global adversary $\mathfrak{A}\_{,g}$ simply analyzes the message flow, local or colluding adversaries $\mathfrak{A}\_{,l}$ or $\mathfrak{A}\_{,c}$ may analyze the existence of pheromones or analyze the routing tables and combine this learned knowledge with knowledge about neighborhoods. The information leakage of leaf positions is caused by subjects that do only join a communication overlay O_t when they are interested in the communication topic t, i.e., being in $\mathcal{S}_t$ or in $\mathcal{R}_t$, or they are required to forward messages, i.e., subjects being in $\mathcal{B}_t$. Obviously, when located in a leaf position, a subject does not have a successor (in case of a recipient) or predecessor (in case of a sender). Thus, such a subject is by definition part of $\mathcal{S}_t$ or $\mathcal{R}_t$. Leaf subjects are contained in the set $\mathcal{L}_t$:

  $$ls(O_t) = |\mathcal{L}_t| = |\{v \in \mathcal{V}_t : d^+(v) = 0\}| \tag{18}$$

  A communication overlay O_t with *fewer* leaf subjects can achieve anonymity with less overhead of anonymization measures.

- *Anonymity Set Size* measures the certainty of an adversary $\mathfrak{A}$ on identifying subjects that are receiving messages, i.e., the adversary $\mathfrak{A}$ tries to discriminate subjects that take part in $\mathcal{R}_t$. For that, the adversary $\mathfrak{A}$ establishes a candidate set $\mathcal{R}_t^*$, which contains the subjects that are potential recipients from the point of view of the adversary $\mathfrak{A}$. The goal of the adversary $\mathfrak{A}$ is to remove subjects from $\mathcal{R}_t^*$ to approximate $\mathcal{R}_t$ using their candidate set $\mathcal{R}_t^*$.

  Initially, $\mathcal{R}_t^*$ is set to $\mathcal{V}$. Depending on the their priori knowledge, the adversary is able to reduce the candidate set at the beginning, i.e., remove subjects from the candidate set $\mathcal{R}_t$ without observing any communication. If the adversary has knowledge about the overlay O_t, they can remove the subjects from the candidate set that are not involved in $\mathcal{V}_t$:

$$\mathcal{R}_t^* \setminus (\mathcal{V} \setminus \mathcal{V}_t) = \mathcal{R}_t^* = \mathcal{V}_t \tag{19}$$

This thesis assumes that the $\mathfrak{A}$ is able to derive subjects involved in a communication overlay O_t; thus, this reduction is always performed. Similar to the adversary in *Leaf Subjects*, the adversary can analyze the message flow or analyze neighborhoods and routing information to learn information about the overlay O_t and its participants $\mathcal{V}_t$.

Naturally, the adversary $\mathfrak{A}$ is also able to remove the set of subjects under their control $\mathfrak{C}_t$ from the candidate set of recipients resulting in $\mathcal{R}_t^* = \mathcal{V}_t \setminus \mathfrak{C}_t$.

The anonymity set size ass is the size of the resulting set $\mathcal{R}_t^*$:

$$ass = |\mathcal{R}_t^*| = |\mathcal{V}_t \setminus \mathfrak{C}_t| \tag{20}$$

- *Anonymity Degree* ad measures the certainty of an adversary $\mathfrak{A}$ of identifying a subject as recipient by evaluating the entropy, i.e., measures the identification probability of recipients $r \in \mathcal{R}_t$. The anonymity degree is computed with the Shannon entropy [131], which requires the probabilities of selecting recipients r when randomly selecting subjects. The probability differs for *inner* subjects and *leaf* subjects and is summarized by Equations (21) and (22). Both first compute the overall probability of selecting a subject of the respective class as recipient which is then normalized to ensure that the sum of probabilities is 1.0.

$$p_{inner} = \frac{|\mathcal{R}_t| - ls(O_t)}{|\mathcal{V}_t| - ls(O_t)} \cdot \frac{1}{|\mathcal{V}_t| - ls(O_t)} \tag{21}$$

$$p_{leaf} = \frac{|\{r \in \mathcal{R}_t : d^+(r) = 0\}|}{ls(O_t)} \cdot \frac{1}{ls(O_t)} \tag{22}$$

The probabilities p_{inner} and p_{leaf} form the probability vector $\mathbf{p_k}$ as provided by Equation (23). The probability vector $\mathbf{p_k}$ is used to compute the Shannon entropy S as provided by Equation (24).

$$\mathbf{p_k} = \left[p_{k_1}, p_{k_2}, \ldots, p_{k_i}, \ldots, p_{k_{|\mathcal{V}_t|}}\right] \tag{23}$$

$$\text{with } p_{k_i} = \begin{cases} p_{leaf}, & \text{if } v_i \in \mathcal{L}_t \text{ and } v_i \in \mathcal{V}_t \\ p_{inner}, & \text{if } v_i \in \mathcal{V}_t \end{cases}$$

$$S = -\sum_{v_i \in \mathcal{V}_t} \left(\mathbf{p_k}[i] \cdot \log_2(\mathbf{p_k}[i])\right) \tag{24}$$

The anonymity degree ad, as provided by Equation (25), is defined as the *normalized* Shannon entropy S with $S_{max} = \log_2(|\mathcal{V}_t|)$ [44].

$$ad = 1 - \frac{S_{max} - S}{S_{max}} = \frac{S}{S_{max}} \tag{25}$$

A larger anonymity degree expresses a higher equality of subjects from the perspective of the adversary $\mathfrak{A}$, which results in a better anonymity.

Table 5 summarizes the evaluation metrics that are used in the following Sections 4.4.5 and 4.4.6 that discuss the improvements of ACO-based communication overlays.

Table 5: Evaluation Metrics

Metric	Explanation
Efficiency:	
$sr_{\mathcal{R}_t}(O_t)$	ratio of recipients that are connected to O_t
$w_{\mathcal{E}}(O_t)$	costs of message dissemination
$cd_{avg}(O_t)$	average latency between $s \in \mathcal{S}_t$ and $r \in \mathcal{R}_t$
Anonymity:	
$ls(O_t)$	number of leaf subjects in O_t
$ass(O_t) \in [0, 1]$	anonymity set size in O_t
$ad(O_t) \in [0, 1]$	anonymity degree of O_t

4.4.3 *Network Configuration*

A thorough evaluation of the proposed novel ACO-based overlay establishment mechanism requires the simulation study to be performed on representative networks. Therefore, this section evaluates variations of the configurations of the network generators for the networks introduced in Section 2.2.4.

Research Question 1 *Which configuration of network size $|\mathcal{V}|$ and network specific parameters is required to create networks that reveal stable graph-theoretic properties according to their model?*

SOCIAL NETWORKS Following [1, 13], the Barabàsi-Albert model establishes networks that are invariant for different settings of m, where m defines the number of connections that are added for every subject that is added to the network. The only, obvious, restriction is $m \leqslant m_0$ as m_0 defines the number of initial subjects from which the social network is grown to its desired size.

In this experiment, the Barabàsi-Albert generator is used with four settings of m to understand the influence of the setting of this parameter on the final network.

$$m = \{2, 3, 4, 5\}$$

Often, smaller networks do not inhibit the typical statistical properties. Therefore, this study also evaluates the properties of networks that are generated with increasing number of subjects $|\mathcal{V}|$.

$$\begin{aligned} |\mathcal{V}| =& \{100; 200; 300; 400; 500; 750; 1,000; 2,000; 3,000, 4,000; \\ & 5,000; 10,000; 25,000; 50,000\} \end{aligned}$$

Each configuration of the Barabàsi-Albert generator is used to create 20 network instance for the evaluation of their graph-theoretic properties; Figure 26 provides an overview of the mean values with their respective 97.5 percentiles.

- *Subject Degree*: Figure 26a visualizes the average subject degree in the generated social networks over increasing number of subjects. As expected, the average degree increases for small network sizes ($\leqslant 1,000$ subjects) and converges quickly to $2 \cdot m$.

The maximum degree is increasing as the preferential attachment ensures that subjects with "more" connections attract additional connections from newly joining subjects.

- *Clustering Coefficient*: The clustering coefficient measures the *ratio of connectedness of the neighbors* of a subject, this value is computed for all subjects and then averaged. Figure 26b visualizes the progression of the average clustering coefficient for increasing number of subjects.

  The preferential attachment causes newly added subjects to favor connections to popular subjects in the backbone. As a result of this, the average neighborhood is sparsely connected; this sparse connectivity in neighborhoods is getting stronger when the network size is increasing.

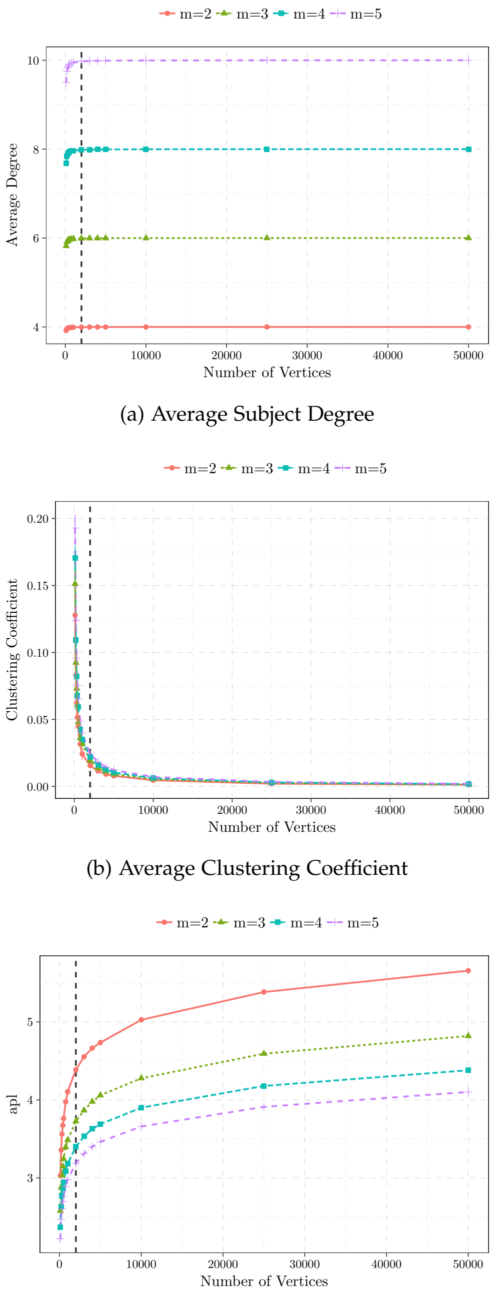
- *Density*: The density of a network describes the ratio of *connections that exist* over the number of *possible connections*. Typically ensures a higher the density of a network shorter path length between any pairs of subjects.

  Social networks provide shortcuts using their backbone while the preferential attachment with low numbers of added connections arrange that the density decreases with increasing number of subjects.

- *Path Length*: The apl is a measure for the average distance between any two subjects in the network. Figure 26c visualizes the apl over increasing number of subjects. It shows that the distances in social networks are only slowly increasing even though the size of the networks is increased heavily at the same time. Also, the larger m, the shorter the distances between any two subjects.

The experiment indicates that a configuration with a network size $|\mathcal{V}|$ of 2,000 subjects and $m = 3$ additional connections with every added subjects is producing representative social networks. The diameter, density, and the maximum degree are, amongst the properties shown in Figure 26 visualized in Appendix B in Figure B.57.

RANDOM NETWORKS Random networks do not reveal a specific structure. Connections between any two subjects are created randomly, only the average number of connections per subject is specified.



(a) Average Subject Degree

(b) Average Clustering Coefficient

(c) Average Shortest Path Length

Figure 26: Graph-theoretic properties of social networks following the Barabàsi-Albert model. The gray, dashed vertical line marks the selected network size of 2,000 subjects.

In this experiment, the Erdős-Rényi generator is used with six different settings of the average number of connections per subject:

$$\overline{d(v)} = \{3, 4, 5, 9, 10, 11\}$$

The network size is varied as in the study of social networks:

$$|\mathcal{V}| = \{100; 200; 300; 400; 500; 750; 1,000; 2,000; 3,000, 4,000; \\ 5,000; 10,000; 25,000; 50,000\}$$

- *Subject Degree*: The average degree is not stable for smaller networks ($|\mathcal{V}| < 1,000$), due to the law of large numbers this deviation from the expected average degree is compensated for larger networks. Without an additional mechanism biasing the probabilities of connection, the maximal subject degree is not increasing as in the generation of social networks. Figure 27a visualizes the average subject degree over increasing network sizes.

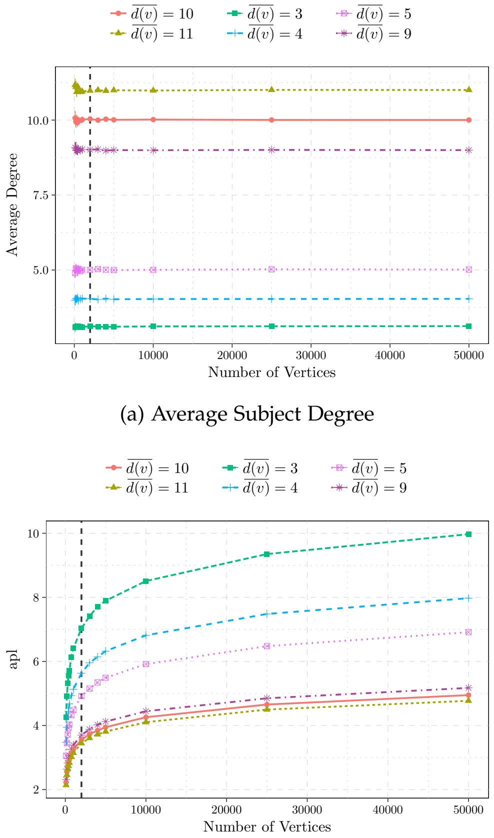
- *Clustering Coefficient & Density*: The number of connections of each subject on the random networks is comparable to the number of on the social networks. As a result of this, the clustering coefficient and density are similar low.

- *Path Length*: Random networks do not provide shortcuts through an inner core. Therefore, the apl and diameter are larger than on social networks. Nonetheless, both path lengths do not grow linearly with the network size. Figure 27b

This experiment indicates that a network size $|\mathcal{V}|$ of 2,000 subjects is sufficient to establish networks that reveal the desired graph-theoretical properties. The setting of the target average subject degree has the expected effect (higher degrees lead to shorter paths, higher density, and higher clustering coefficient), yet, the changes in graph-theoretic properties are proportional and not surprising. Figure B.58 in Appendix B visualizes the results of this experiment, i.e., shows the graph-theoretic properties of random networks over increasing number of subjects.

As these random networks may be established using a gossiping mechanism like SCAMP [59], the following evaluation considers two settings of the average number of connections per subjects:

- $\overline{d(v)} = 10$: the evaluation of the SCAMP mechanism shows that the average degree of subjects in larger networks ($|\mathcal{V}| > 50,000$) is larger than 10. These random networks are referred to by random$_{10}$.



(a) Average Subject Degree

(b) Average Shortest Path Length

Figure 27: Graph-theoretic properties of random networks following the Erdős-Rényi model. The gray, dashed vertical line marks the selected network size of 2,000 subjects.

- $\overline{d(v)} = 4$: the average degree of subjects in SCAMP is given by $\log(|V|)$. Therefore, the average degree increases with additional subjects in the communication system, an average degree of 4 is being used as an estimate for the lower border. These random networks are referred to by random$_4$.

RGG RGGs inherit their structure from random placement of added subjects and a distance-based connection structure. As the plane on which subjects are placed is limited, an increasing number of subjects lead to a denser network with more connections. Nonetheless, graph-theoretic properties inspect a global perspective on the network while the increasing number of subjects and connections have a local effect. As soon as an area of the location-plane is covered by a subject, adding additional subjects to this area does not change graph-theoretic properties.

In this study, three different settings of the connection distance threshold of a subject are used:

$$\mathtt{threshold} = \{0.1, 0.15, 0.2\}$$

The network is fixed to a 2-dimensional plane ($d = 2$). The network size is chosen similar to the study of social networks:

$$\begin{aligned}|\mathcal{V}| = &\{100; 200; 300; 400; 500; 750; 1,000; 2,000; 3,000, 4,000;\\ &5,000; 10,000; 25,000\}\end{aligned}$$

- *Subject Degree*: The degrees of the subjects in RGGs depends largely on the number of subjects, the connection distance threshold, and the dimension of the network. Therefore, both the average and the maximal degree of subjects increase linearly with increasing number of subjects.

- *Clustering Coefficient & Density*: The probability of connecting any two subjects does not depend on the number of subjects in the network but on their random location. As a result of this, clustering coefficient and density are stable with an increasing number of subjects. The clustering coefficient, however, is unstable for a smaller number of subjects ($|\mathcal{V}| \leqslant 2,000$).

- *Path Length*: The properties of path lengths, apl and diameter, are also depending on the distance threshold and the dimension of the network. If there are enough subjects such that every location is covered in the connection threshold of any subject, the apl and diameter are stable and do not change with additional subjects.

The study indicates that a network size $|\mathcal{V}|$ of 2,000 subjects is sufficient to establish networks revealing stable graph-theoretical properties. Adding additional subjects does not change graph-theoretical properties. Figure B.59 in Appendix B visualizes the progression of the graph-theoretic properties with varied network size and varied connection distance threshold.

SUMMARY The performed study of graph-theoretic properties shows that the following configurations of the network generators establish representative network instances that can be used to thoroughly evaluate the overlay establishment mechanisms. Table 6 summarizes the selected configurations.

Table 6: Selected Network Configuration

Parameter	Instantiation
Social Network:	
Network size $\lvert\mathcal{V}\rvert$	2,000
Connections per added subject m	3
Random Network:	
Network size $\lvert\mathcal{V}\rvert$	2,000
Target average degree $\overline{d(v)}$	10 (random_{10}) 4 (random_4)
RGG:	
Network size $\lvert\mathcal{V}\rvert$	2,000
Dimension d	2
Threshold t	0.1

4.4.4 *ACO-Configuration*

The ACO-based overlay establishment mechanism offers several parameters that control the ability of the optimization heuristic to establish a good—sometimes close to optimal—communication overlay. This section analyzes the effect of the individual parameters and concludes with a recommended configuration.

In this evaluation, an experiment varies one of the parameters while the others remain unchanged; the success ratio $sr_{\mathcal{R}_t}(O_t)$ and the weight $w_{\mathcal{E}}(O_t)$ are analyzed to reveal beneficial—and detrimental—parameter configurations. Each experiment follows the simulation setup presented by Section 4.4.1; the networks are configured as shown in Section B, however, only the random_4 instance is being used to perform this experiment.

4.4.4.1 *Number of Agents*

The number of agents is a core parameter of the ACO heuristic that trades convergence speed and probability against computational complexity. Considering an anonymous communication system with heterogeneous subjects, it is essential to establish the communication overlays O_t without overloading the system with agents.

Research Question 2 *How many agents (noa) need to be emitted to establish a communication overlay O_t that connects the sender with all recipients and exhibits a lower weight than traditional flooding-based communication overlays?*

The number of agents noa is varied between $\{2, 5, 10, 20, 50, 100, 150\}$ to identify the number of agents that are necessary to fulfill the given challenge.

The experiment shows that the ACO-based overlay establishment mechanism is independent from the number of agents able to create communication overlays with $sr_{\mathcal{R}_t} = 1.0$. Figure 28 visualizes the weight of the resulting communication overlays $w_{\mathcal{E}}(O_t)$ of the varied ACO configurations and added the baselines of the flooding-based communication overlay as well as the Steiner tree approximation StA. As such, Figure 28 shows that the overlays that are established with a lower number of agents, for example, $noa = \{2, 5\}$, are not improving the weight in comparison the flooding-based communication overlays. The communication overlays that are established with $noa \geqslant 10$ are improving the weight of the communication overlay $w_{\mathcal{E}}(O_t)$ by $\approx 10\%$. The experiment also shows that increasing the number of agents beyond 10–20 does not yield a significant reduction of the communication overlay O_t.

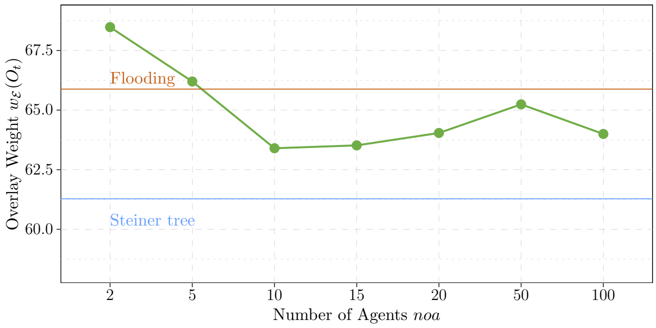


Figure 28: Varying the number of emitted agents (noa) and its influence on the weight of the communication overlay ($w_{\mathcal{E}}(O_t)$).

4.4.4.2 *Recipient Ratio*

The recipient ratio rr is studied to analyze whether the first impression of improvement weight improvement over the state of the art mechanism to establish communication overlays O_t is limited to a specific ratio of recipients.

Research Question 3 *Is the weight improvement of the ACO-based communication overlays depending on the ratio of recipients* rr*?*

A study is performed to see whether or not the ACO-based overlay establishment is able to create efficient communication overlays with different recipient ratios. The number of recipients is drawn from a Gaussian distribution with the mean $\mu = |\mathcal{V}_t| \cdot \mathrm{rr}$ and $\mathrm{sd} = 2$, and the ratio of recipients rr is varied:

$$\mathrm{rr} \in \{0.015, 0.02, 0.05, 0.1, 0.2, 0.5\}$$

The ACO-based overlay establishment mechanism is able to establish fully connected communication overlays ($\mathrm{sr}_{\mathcal{R}_t} = 1.0$) without being influenced by changes in the recipient ratio.

The novel ACO-based overlay establishment mechanism creates communication overlays that are using less connections than traditional flooding-based shortest path overlays. In the worst case, the ACO-based communication overlays contains close to equal numbers of connections, i.e., similar weight $w_{\mathcal{R}}(O_t)$, as the traditional, flooding-based communication overlays; in better cases the ACO-based communication overlays reduce the weight in comparison to traditional, flooding-based communication overlays by up 8.4%. Table 7 summarizes the respective overlay weights depending on the varied recipient ratio rr and the relative weight improvement.

Table 7: Overlay weight $w_{\mathcal{E}}(O_t)$ with varying ratio of recipients rr.

rr	ACO	flooding	rel. weight improvement
0.015	64.66667	65.36667	1.0704
0.02	79.56667	82.63333	3.7112
0.05	161.89655	175.10345	7.5423
0.1	276.72000	302.12000	8.4073
0.2	471.68000	513.32000	8.1119
0.5	901.36000	956.20000	5.7352

Figure 29 visualizes the weights for the communication overlays over the varying recipient ratios rr and shows that the ACO-based overlays are at least as efficient as the flooding-based overlays.

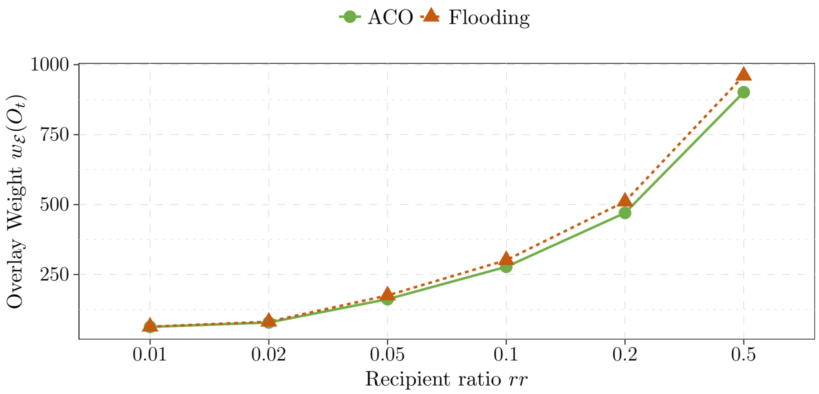


Figure 29: Overlay weight $w_{\mathcal{E}}(O_t)$ with varying ratio of recipients rr.

4.4.4.3 *Time to Live of Agents*

The increasing TTL for agents is required to ensure that all recipients may find the communication overlay O_t, or the sender $s \in \mathcal{S}_t$, and the system is not overloaded with roaming agents. For that, a configuration is required that limits the path length in the communication overlay while it still enables to ACO-based optimization, which leads to slightly longer paths.

Research Question 4 *Is the variable TTL configuration enabling the ACO-based overlay establishment mechanism to establish connected and improved communication overlays?*

To understand the influence of varying TTL settings, the ACO-based overlay establishment mechanism establishes overlays with the following TTLs:

$$ttl \in \{10, 25, 50, 75, 100, 125\}$$

Setting the TTL to 10, the ACO-based overlay mechanism is not able to establish a communication overlay with $sr_{\mathcal{R}_t} = 1.0$ on random networks; The $ttl = 10$ TTL setting enables only approximately 64% of recipients to connect to the sender. This result is expected as the mean diameter of a random network and 2,000 subjects is already 12.5. All other TTL settings are able to successfully establish communication overlays with $sr_{\mathcal{R}_t} = 1.0$.

Figure 30 and 31 reveals two observations. In Figure 30, it is shown that larger TTL values ($\geqslant 75$) lead to increasing overlay weights on social networks eventually. This behavior is caused by a "misleading" optimization where paths are aggregated even

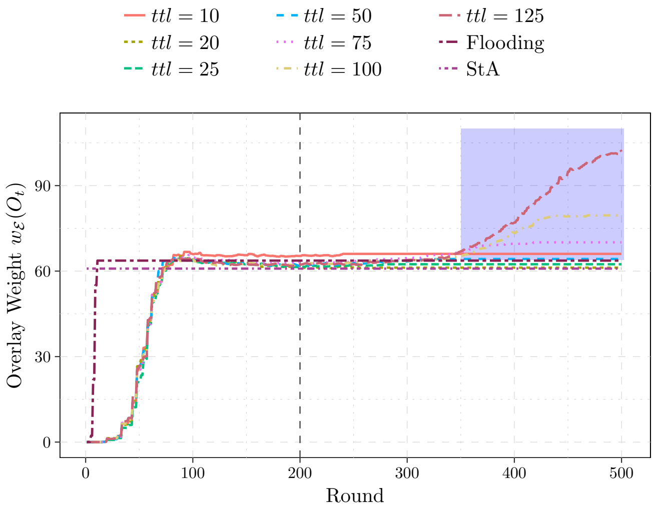
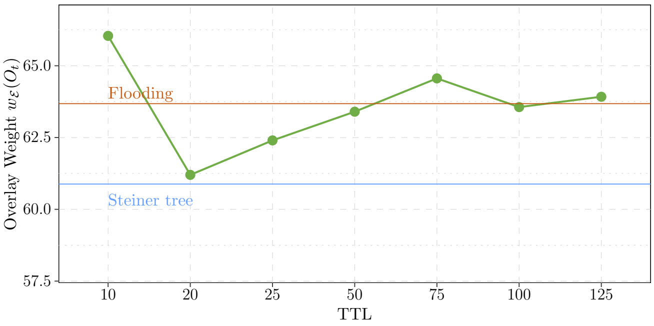


Figure 30: Weight $w_{\mathcal{E}}(O_t)$ for varied TTL values on social networks. The blue highlight shows "misleading" optimization. After the dashed line round 200, no additional agents are emitted by the recipients.

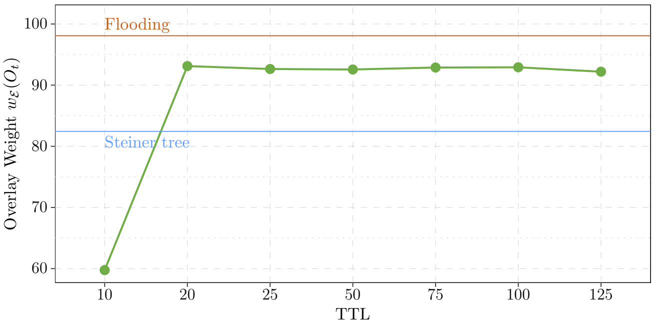
though they become unnecessarily long, for example, by connecting distinct subtrees using longer detours. Smaller TTL values $(20, 25)$, which are in the range of approximately 4 times the diameter, provide more efficient overlays that even converge towards the Steiner tree approximation StA. Figure 31a visualizes the overlay weight for varied TTL values on social networks before the misleaded optimization takes place. It also confirms the conclusion that the TTL is best configured in the range of approximately 4 times the diameter; larger TTL values increase the overlay weight $w_{\mathcal{E}}(O_t)$ again. Figure 31b visualizes the respective overlay weights for varied TTL values on random networks; here, the "misleading" optimization on random networks is not shown. All evaluated TTL values larger than 10 enable the ACO-based overlay establishment mechanism to create communication overlays with less weight than the flooding-based shortest path communication overlays.

4.4.4.4 *Diffusion*

The diffusion parameter controls the amount of *new* pheromones that subjects spread to adjacent connections. Without diffusion, agents have to be directly on a subject that is incident to a connection with pheromones; the path aggregation, and therefore the resulting overlay optimization, is harder. On the other side,



(a) Weight $w_{\mathcal{E}}(O_t)$ on social networks.



(b) Weight $w_{\mathcal{E}}(O_t)$ with varied TTL values on random networks.

Figure 31: Varying the TTL of emitted agents on social networks (a) and on random networks (b). The $\text{ttl} = 10$ configurations are included for completeness, even though this settings are to small to establish a connected overlay.

if diffusion spreads the full amount of new pheromones, all incident connections of a subject are likely to have a similar amount of pheromones. With that, the activation of an optimized overlay becomes a matter of luck as the activation agents (see Phase 3 in Section 4.3.5) may transition into dead ends or towards unprofitable paths.

Research Question 5 *Is the ACO-inherent optimization functional with the limited 1-hop diffusion mechanism?*

A study varies the diffusion parameter and evaluates the resulting success ratio and weight of the communication overlay. The diffusion is configured with the following factors to spread different ratios of pheromones to incident connections:

$$d_{1h} \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$$

A diffusion value $d_{1h} \geqslant 0.75$ prevents the ACO-based overlay establishment mechanism from creating communication overlays with $sr_{\mathcal{R}_t} = 1.0$. With a diffusion ratio $d_{1h} = 0.75$, the ACO-based overlay establishment is not able to connect more than 50% of recipients; a diffusion ratio $d_{1h} = 0.75$ occasionally misses to connect one recipient.

Figure 32 shows the weight of the established overlays $w_{\mathcal{E}}(O_t)$ for varied diffusion settings d_{1h} over time. Despite establishing overlay successfully with no diffusion ($d_{1h} = 0.0$), the optimization of the ACO-based overlay establishment is difficult such that the established overlays utilize more connections than necessary. In between the extreme values ($0.0 \ll d_{1h} \ll 1.0$), the one hop diffusion supports ACO in the overlay optimization.

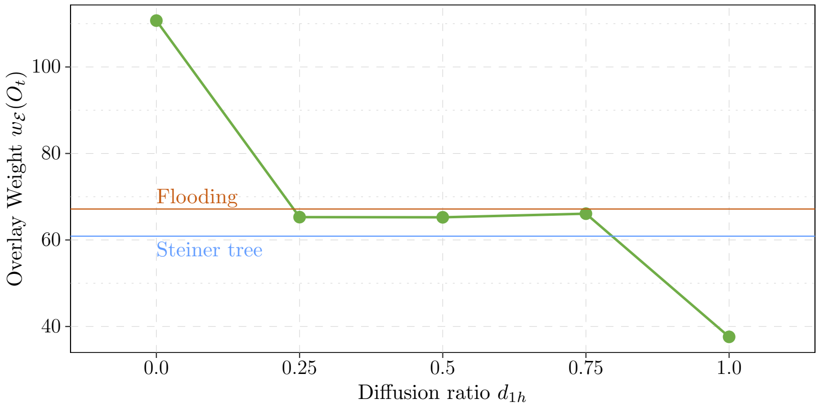


Figure 32: Weight $w_{\mathcal{E}}(O_t)$ for varied diffusion factors. The extreme values inhibit the establishment of optimized overlays; for $d_{1h} = 1.0$, the overlay establishment is not successful.

4.4.4.5 *Evaporation*

The evaporation γ enables the optimization of the communication overlays by removing old pheromones over time. However, removing pheromones too fast may lead to a more difficult overlay establishment and optimization.

Research Question 6 *In which order of magnitude has the evaporation γ to be configured to enable the ACO-based overlay establishment mechanism to exploit pheromones* and *to optimize the communication overlay?*

A study varies the evaporation γ to evaluate the established communication overlays with respect to their success ratio $sr_{\mathcal{R}_t} = 1.0$ and their weight $w_{\mathcal{E}}(O_t)$:

$$\gamma \in \{0.0, 0.01, 0.05, 0.07, 0.1, 0.25, 0.5, 0.75, 1.0\}$$

With $\gamma = 1.0$, the evaporation removes all pheromones in every round. Without pheromones on connections, the ACO-based overlay mechanism degenerates to parallel random walks and is no longer able to consistently establish connected communication overlays with $sr_{\mathcal{R}_t} = 1.0$.

High evaporation factors ($\gamma > 0.1$) remove larger parts of the pheromones and result in overlays with a higher weight $w_{\mathcal{E}}(O_t)$; as larger parts of the pheromones dissolve in a short time, and the agents of ACO loose the ability to exploit previously established trails. Figure 33 shows that evaporation factors $\gamma \leqslant 0.1$ facilitate the optimization of the established communication overlays.

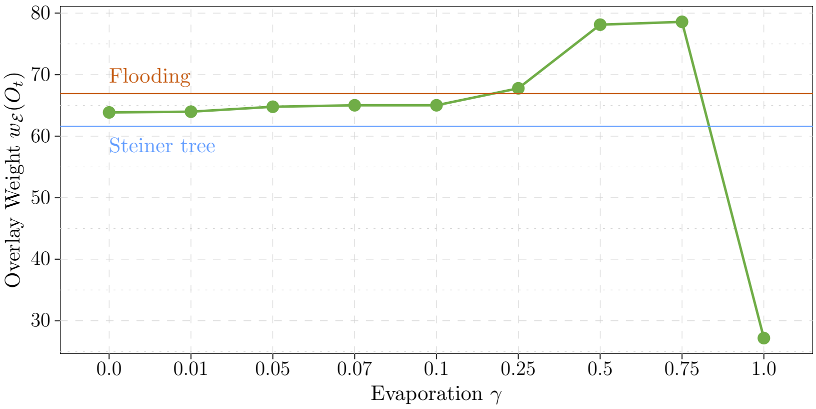


Figure 33: Weight $w_{\mathcal{E}}(O_t)$ with varied the factor of proportional evaporation γ. The $\gamma = 1.0$ setting is not able to establish connected overlays reliably in 500 rounds and is included for completeness.

4.4.4.6 *Strictness*

The strictness σ of agents controls the relevance of pheromones for the random walk of agents; the recipients initiate the strictness of agents with a random value (see Section 4.3.3). The window of strictness values from which the recipients draw the strictness value is capped with the provided values $[\sigma_{low}, \sigma_{up}]$. Lower strictness values are expected to worsen the optimization abilities as pheromones have a lower influence on the random walk of agents.

Research Question 7 *Which window of strictness values* $[\sigma_{low}, \sigma_{up}]$ *provides the ACO-based overlay establishment mechanism with a mixture of agents that exploit pheromones and explore the system to optimize the communication overlay?*

Both strictness limits σ_{up} and σ_{low} are varied in a study to evaluate the established communication overlays with respect to their success ratio $sr_{\mathcal{R}_t} = 1.0$ and their weight $w_{\mathcal{E}}(O_t)$:

$$\sigma_{\mathrm{up}} \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$$
$$\sigma_{\mathrm{low}} \in \{0.0, 0.25, 0.5, 0.75, 0.9, 1.0\}$$

Obviously, simulations with $\sigma_{\mathrm{low}} > \sigma_{\mathrm{up}}$ are skipped.

All strictness windows $[\sigma_{\mathrm{low}}, \sigma_{\mathrm{up}}]$ enable the agents to successfully establish the communication overlays. Figure 34 visualizes the overlay weights $w_{\mathcal{E}}(O_t)$ over varied strictness intervals. For that, the x-axis represents the upper strictness limits σ_{up} while the values represent the lower strictness limits σ_{low}.

However, Figures 34a and 34b show that low upper limits prevent the ACO-based overlay establishment from successfully optimizing the created overlays, resulting in a higher weight of the communication overlays. A setting of $\sigma_{\mathrm{up}} = 1.0$ yields the best optimization as it enables recipients to establish a mixture of exploring and exploiting agents.

The lower limit σ_{low} enables the agents to exploit the structure by highlighting already known paths. On social networks, the exploitation of the backbone is essential for the optimization. Figure 34a shows that a lower limit $\sigma_{\mathrm{low}} \geqslant 0.75$ yields the best results. Figure 34b visualizes that random networks do not benefit from changing the lower limit σ_{low}.

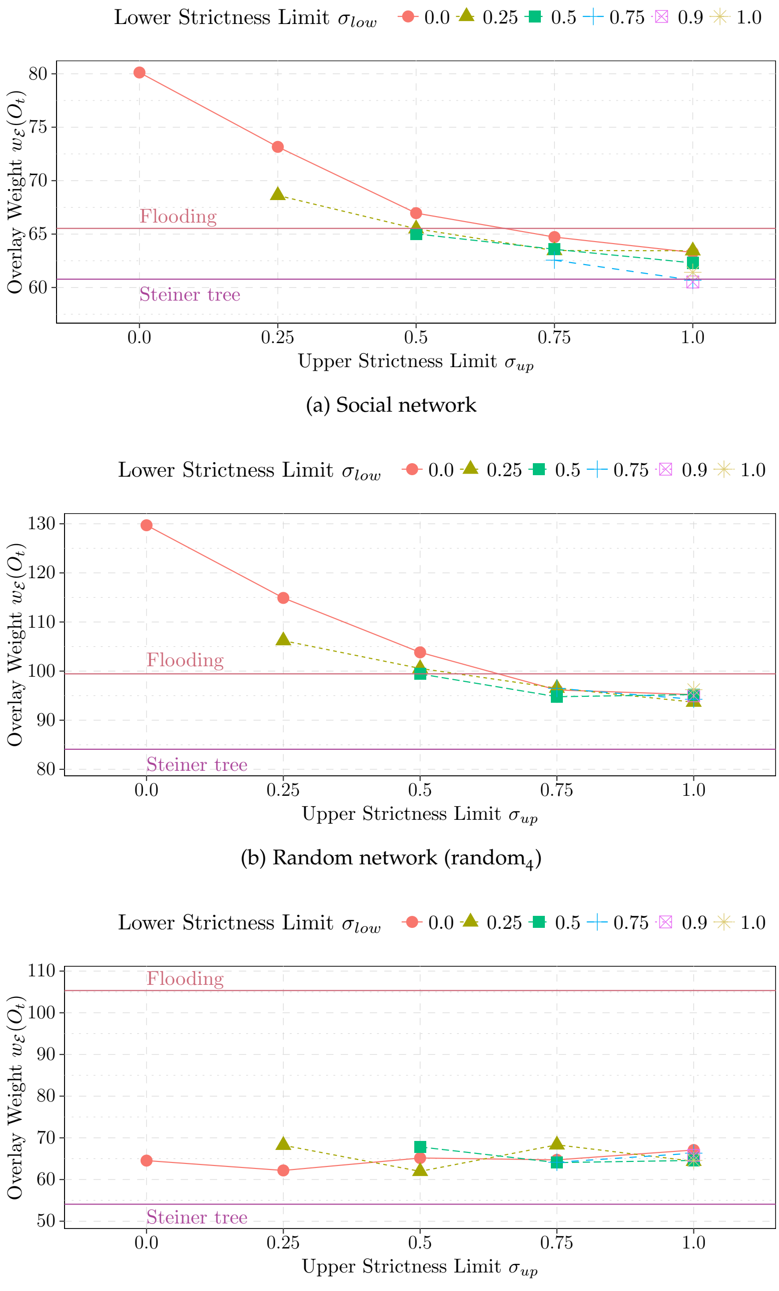
RGGs, however, do not benefit from specific strictness σ settings. The resulting changes are negligible compared to the already achieved optimization. Figure 34c visualizes the comparably stable overlay weights $w_{\mathcal{E}}(O_t)$ for varied strictness σ intervals.

4.4.4.7 *Summary*

The studies performed in this section reveal the influence of several ACO parameters on the ACO-based mechanism to establish optimized overlays. Table 8 summarizes the best performing parameter settings for the different networks that are also used for the following evaluation.

4.4.5 *Efficiency Improvements*

This section recapitulates and evaluates the quantitative simulation with respect to the communication overlay properties con-



(a) Social network

(b) Random network (random$_4$)

(c) RGG

Figure 34: Overlay weight $w_{\mathcal{E}}(O_t)$ with varied strictness values σ of agents.

cerning efficiency. For that, the key lessons of the evaluations of the ACO parametrization in the previous Section 4.4.4 are reconsidered under the light of the following challenges and research questions:

Table 8: Best-performing ACO Parameters

Network	NoA	TTL	Diffusion	Evap. (γ)	Strictness (σ)
Social	20	25	0.5	0.07	$[0.9, 1.0]$
Random_{10}	25	25	0.25	0.07	$[0.0, 1.0]$
Random_{4}	25	50	0.25	0.07	$[0.5, 1.0]$
RGG	50	50	0.25	0.01	$[0.5, 1.0]$

- Are ACO-based communication overlays affecting the ability of recipients to communicate, i.e., is ACO-based overlay establishment negatively affecting the receiver success ratio $sr_{\mathcal{R}_t}(O_t)$?
- Is ACO able to exploit the topology of the logical underlay G to improve the communication overhead, i.e., to reduce $w_{\mathcal{E}}(O_t)$ in comparison to flooding-based shortest path communication overlays?
- To what extent is the communication delay $cd_{avg}(O_t)$ affected the refrain from using shortest paths for overlay establishment?

The experiments in this evaluation follow the simulation setup described in Section 4.4.1, the ACO mechanism is initialized with the best performing setup which summarized in 8. Where applicable, the conclusions of the previous evaluation (ACO-Configuration in Section 4.4.4) are taken into account in the discussion.

4.4.5.1 *Receiver Success Ratio* – $sr_{\mathcal{R}_t}(O_t)$

The analysis of the receiver success ratio $sr_{\mathcal{R}_t}(O_t)$ addresses the first research question of this section:

Research Question 8 *Are ACO-based communication overlays affecting the ability of recipients to communicate, i.e., is ACO-based overlay establishment negatively affecting the receiver success ratio $sr_{\mathcal{R}_t}(O_t)$?*

As shown in the evaluation of the ACO-parameters, the novel ACO-based overlay establishment mechanism can always achieve $sr_{\mathcal{R}_t}(O_t) = 1.0$. However, the configuration of the ACO overlay establishment mechanism has to fulfill a set of requirements for that:

- The TTL has to be larger than the diameter of the logical underlay G. An appropriate setting is 3–5 times the diameter. A smaller TTL may be chosen if the communication group is known to be "closer" together, i.e., the distance between sender and recipients is known to be smaller than the TTL.

- The evaporation factor γ has to be configured such that the pheromone trails guide the agents through the system without forcing them into long detours. Also, the less structure the logical underlay G reveals, the lower the evaporation has to be, for example, a social network enables agents to utilize the inner core to establish the connectivity between sender and recipients, while a random network does not support the agents with a "known" and exploitable structure.

- The diffusion should spread around 50% of the gained pheromones support the optimization. Lower values impede the optimization; larger values impede the overlay establishment.

However, small TTL limits ($ttl < 10$) may limit the ability of the ACO mechanism to successfully connect all recipients; similarly do extreme values of diffusion ($d_{1h} \to 0.0$ and $d_{1h} \to 1.0$), and larger values of evaporation ($\gamma > 0.1$) impede the establishment of connected communication overlays.

4.4.5.2 *Communication Costs – $w_{\mathcal{E}}(O_t)$*

The analysis of the communication costs, measured by the weight of the communication overlay $w_{\mathcal{E}}(O_t)$, addresses the second research question of Section 4.4.5:

Research Question 9 *Is ACO able to exploit the topology of the logical underlay* G *to improve the communication overhead, i.e., to reduce* $w_{\mathcal{E}}(O_t)$ *in comparison to flooding-based shortest path communication overlays?*

As the Pub/Sub-based communication model enables native group communication, i.e., the message duplication is pushed as close to the recipients as possible, the measuring of the communication costs with the weight $w_{\mathcal{E}}(O_t)$ is in line with the efficiency definition in Section 2.1.1.

The novel ACO-based communication overlays are improving the weight $w_{\mathcal{E}}(O_t)$ and approach the Steiner tree approximation StA (see Section 2.2.5 for more details) in many cases. Figure 35 shows the aggregated mean weights $w_{\mathcal{E}}(O_t)$. The bars are grouped on the x-axis by the network (social, random$_{10}$, random$_4$, RGG); each bar represents the weight $w_{\mathcal{E}}(O_t)$ of either the ACO-based communication overlay, the flooding-based shortest path communication overlay, or the StA.

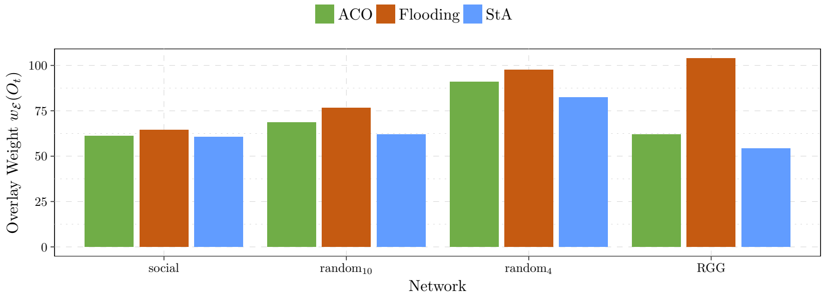


Figure 35: Weight $w_{\mathcal{E}}(O_t)$ of the established communication Overlays. The network types are arranged at the x-axis, the mean weight of the created communication overlays is placed at the y-axis—a lower weight $w_{\mathcal{E}}(O_t)$ is better.

SOCIAL NETWORKS The usage of the ACO-based overlay establishment mechanism enables to utilize the densely connected core to the advantage of the path aggregation and therefore to the improvement of the weight $w_{\mathcal{E}}(O_t)$.

The flooding-based shortest path communication overlay strictly favors shortest paths. Social networks facilitate this behavior as they enable flooding-based communication overlays to exploit the densely connected core and benefit from path aggregation as well.

The path consolidation of ACO reduces the weight $w_{\mathcal{E}}(O_t)$ by 5.69% compared to the flooding-based overlay and converges towards the weight of the optimal solution of the Steiner tree approximation StA. Table 9 reports the respective absolute overlay weights $w_{\mathcal{E}}(O_t)$ of the overlay and the relative differences to the StA.

RANDOM NETWORK In random networks, the ACO-based overlay establishment mechanism cannot operate on a densely connected core where many paths cumulate. Without revealing

Table 9: Weights $w_{\mathcal{E}}(O_t)$ of O_t on social networks.

	ACO	Flooding	StA
$w_{\mathcal{E}}(O_t)$	62.2	65.95	62
rel. to StA	+0.3%	+6.4%	

such a structure, it is more difficult to aggregate message dissemination paths and to reduce the weight $w_{\mathcal{E}}(O_t)$ of the communication overlay.

The focus of the flooding-based overlay establishment on utilizing only shortest paths leads to a higher weight in structureless random networks; the flooding-based overlay mechanism cannot benefit from the implicit path consolidation without a densely connected core where shortest paths are cumulating.

The path consolidation of ACO reduces the weight $w_{\mathcal{E}}(O_t)$ by 10.61% compared to the flooding-based overlay on the random_{10} networks, where subjects have an average degree of 10. On the sparser random_4 networks, with an average subject degree of 4, the improvement of the weight $w_{\mathcal{E}}(O_t)$ is 6.71%. The difference is explained by the introduction of additional connections in the random_{10} networks; the ACO-based overlay establishment mechanism exploits these additional connections and improves the weight $w_{\mathcal{E}}(O_t)$ by additional 4%. In comparison to the Steiner tree approximation StA, ACO-based communication overlays nearly half the overhead of flooding-based communication overlays.

Table 10 reports the respective absolute overlay weights $w_{\mathcal{E}}(O_t)$ of the overlay and the relative differences to the StA.

Table 10: Weights $w_{\mathcal{E}}(O_t)$ of communication overlays O_t on both random_{10} and random_4 networks.

	ACO	Flooding	StA
random_{10}:			
$w_{\mathcal{E}}(O_t)$	68.47	76.6	59.30
rel. to StA	+15.46%	+29.17%	
random_4:			
$w_{\mathcal{E}}(O_t)$	90.93	97.47	82.48
rel. to StA	+10.24%	+18.17%	

RANDOM GEOMETRIC GRAPH (RGG) The ACO-based overlay establishment mechanism is able to exploit the parallel structure of connections that is caused by the locality-based connection structure in RGGs. With that parallel structure, the ACO-based overlay establishment mechanism can easily consolidate paths and reduce the weight of the communication overlay $w_{\mathcal{E}}(O_t)$.

Flooding-based communication overlays may have multiple shortest paths to chose from and select one randomly. With that, the weight of the communication is often unnecessarily high.

The ACO-based overlay establishment mechanism is able to reduce the weight $w_{\mathcal{E}}(O_t)$ by 40.66% compared to flooding-based shortest path communication overlays. Table 11 reports the respective absolute overlay weights $w_{\mathcal{E}}(O_t)$ of the overlay and the relative differences to the StA.

Table 11: Weights $w_{\mathcal{E}}(O_t)$ of O_t on RGG networks.

	ACO	Flooding	StA
$w_{\mathcal{E}}(O_t)$	59.83	100.82	52.33
rel. to StA	+14.3%	+92.7%	

4.4.5.3 *Communication Delay* – $cd_{avg}(O_t)$

The communication delay is estimated by the length of the path between sender $s \in \mathcal{S}_t$ and the respective recipients $r \in \mathcal{R}_t$.

The analysis in this section addresses the third research question of Section 4.4.5:

Research Question 10 *To what extent is the communication delay* $cd_{avg}(O_t)$ *affected the refrain from using shortest paths for overlay establishment?*

By aggregating dissemination paths and refraining from strictly favoring shortest paths, it is expected that the average distance between sender and recipients increases when creating the communication overlays with the ACO-based overlay establishment mechanism.

SOCIAL NETWORK The average distance between sender and recipients on social networks increases by 10.42% from 4.8 hops to 5.3 hops when using ACO-based communication overlays instead of the flooding-based shortest paths communication overlays.

By exploiting the inner core of the logical underlay, the extension of the average distance is usually no more than one additional hop. With that, the increase of the communication delay by using the ACO-based overlay establishment is only marginal.

RANDOM NETWORK The average distance between sender and recipients on a random network is larger due to the lack of the inner core providing shortcuts between the subjects. The extension of the paths is, however, only 8.19% from 4.52 hops to 4.89 hops on random networks with an average degree of 10 when using the ACO-based communication overlays instead of the flooding-based communication overlays. On random networks with average degree of 4, the average distances increases by 4.53% from 5.96 hops to 6.23.

RGG The distance in RGGs is determined by their (random) location and is limited by dimension and connection distance threshold. The path extension that is caused by the usage of ACO-based communication overlays is on average 26.47% from 6.8 to 8.6 hops.

Figure 36 and Table 12 report the respective absolute results for the communication delay $cd_{avg}(O_t)$ and the relative distance of the delay $cd_{avg}(O_t)$ between the ACO-based overlays and the flooding-based overlays. While the relative growth of the communication delay seems to outweigh the efficiency improvement, the absolute delay growth is only between 0–1 additional hop for social and random networks and 1–2 additional hops for RGGs.

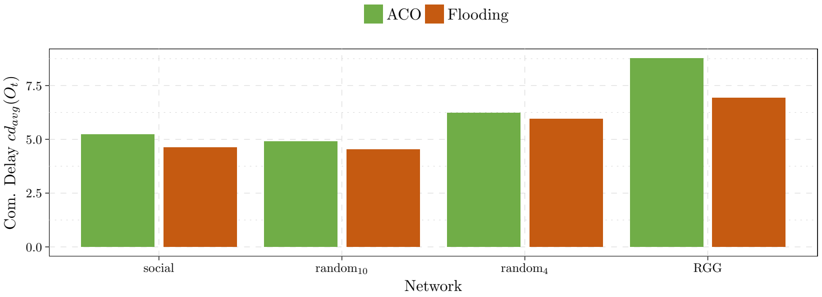


Figure 36: Communication delay $cd_{avg}(O_t)$ of the established communication Overlays. The network types are arranged at the x-Axis, the mean weight of the created communication overlays is placed at the y-Axis—a lower communication delay $cd_{avg}(O_t)$ is better.

Table 12: Communication delay $\mathrm{cd}_{avg}(O_t)$ of ACO-based and flooding-based overlays and the relative difference of the communication delay $\mathrm{cd}_{avg}(O_t)$.

Network	ACO	Flooding	rel. Difference
Social	5.3	4.8	10.42%
Random_{10}	4.89	4.52	8.19%
Random_4	6.23	5.96	4.53%
RGG	8.6	6.8	26.47%

4.4.5.4 *Summary*

The novel ACO-based overlay establishment mechanism succeeds in all three research questions by providing fully connected communication overlays $sr_{\mathcal{R}_t}(O_t) = 1.0$ (*RQ 8*) with a reduced weight $w_{\mathcal{E}}(O_t)$ (*RQ 9*) and only slightly increasing distance between sender and recipients $\mathrm{cd}_{avg}(O_t)$ (*RQ 10*). Table 13 summarizes the results of the performed simulation study.

Table 13: ACO-based communication overlays compared to flooding-based shortest path communication overlays.

Network	$sr_{\mathcal{R}_t}(O_t)$	$w_{\mathcal{E}}(O_t)$	$\mathrm{cd}_{avg}(O_t)$
Social	1.0	62.2 (-5.69%)	5.3 (+10.42%)
Random_{10}	1.0	68.47 (-10.61%)	4.89 (+8.19%)
Random_4	1.0	90.93 (-6.71%)	6.23 (+4.53%)
RGG	1.0	59.83 (-40.66%)	8.6 (+26.47%)

4.4.6 *Maintaining Anonymity while Improving Efficiency*

The evaluation above was focussed on the efficiency improvement of the overlays that are constructed with the novel ACO-based overlay establishment mechanism. This section will complement this by evaluating the anonymity of overlays that are established with the ACO mechanism.

The following challenges and research questions are answered:

- Is the novel ACO-based overlay establishment affecting anonymity with its efficiency improvements?

- What are the costs of the achieved anonymity?

The analysis in this section focusses on the anonymity of recipients. Sender anonymity will be discussed in Chapter 6, the Sections 6.1 and 6.3 substantiate this differentiation.

4.4.6.1 *Anonymity Set Size* – $\mathfrak{ass}$

The anonymity set size $\mathfrak{ass}$ (see Section 2.4.1 and 4.4.2) measures anonymity as the size of the set in which the recipients can hide. That is the set in which the adversary $\mathfrak{A}$ cannot discriminate any member. This analysis studies the first research question of this section:

Research Question 11 *Is the ACO-based overlay establishment impeding anonymity with its efficiency improvements?*

Facing the service provider as global passive adversary $\mathfrak{A}_{p,g}$, the anonymity set size $\mathfrak{ass}$ equals the overlay size $|\mathcal{V}_t|$ as the adversary $\mathfrak{A}_{p,g}$ can remove all subjects $v \notin \mathcal{V}_t$ from the anonymity set. The ACO-based overlay establishment consolidates overlay paths, i.e., it reduces the number of connections in the communication overlay. As a result of this, the ACO mechanism reduces the number of brokers $|\mathcal{B}_t|$ in the communication overlay in the same order of magnitude as it removes connections.

On *social networks*, the ACO-based communication overlay decreases the mean number of subjects by 5.60% to 61.70 compared to 65.36 on flooding-based shortest path communication overlays. On *random networks* with an average degree of 10, the ACO-based communication overlay mechanism removes only 10.57% of the brokers and reduces the number of subjects to 69.4 compared to 77.6 on flooding-based communication overlays (on random networks with degree 4, the reduction is 7.06% from 97.92 to 91.01). The connection structure on *RGGs* enables an immense efficiency improvement, as a result of this, the mean number of subjects on ACO-based communication overlays is dropping by 40.24% to 62.74 compared to 104.98 on flooding-based communication overlays.

The anonymity set size $\mathfrak{ass}$ provides an intuitive anonymity measurement and is reported for the networks analyzed in the previous evaluation (Section 4.4.5) in Table 14. However, the anonymity set size loses its expressiveness if the identification probabilities of the subjects in the anonymity set are not

Table 14: The average anonymity set size $\mathfrak{ass}$ of ACO-based communication overlays compared to the average anonymity set size $\mathfrak{ass}$ flooding-based communication overlays.

Network	ACO	Flooding	rel. Difference
Social	61.7	65.36	-5.60%
Random_{10}	69.4	77.6	-10.57%
Random_4	91.01	97.92	-7.06%
RGG	62.74	104.98	-40.24%

equal. The following section, therefore, analyzes the identification probabilities—and shows that not all identification probabilities are equal—and takes a different approach towards the anonymity quantification.

4.4.6.2 *Entropy-based Anonymity Evaluation*

The global passive adversary $\mathfrak{A}_{p,g}$ is able to derive the communication overlay by tracing the messages through the communication overlay O_t. Moreover, the $\mathfrak{A}_{p,g}$ can identify subjects v in leaf positions $d^+(v) = 0)$ as recipients $v = r \in \mathcal{R}_t$ by utilizing knowledge about the communication model.

With the ability to identify leaf subjects as recipients, the probability of identification is not equal for all subjects in $\mathcal{V}_t$; therefore, the expressiveness of the anonymity set size $\mathfrak{ass}$ is limited. In the following paragraphs, it is argued why it is realistic that the adversary $\mathfrak{A}_{p,g}$ is able to identify leaf subjects as recipients, how many subjects are exposed by this property, how this problem can be resolved, and the resulting anonymity protection using entropy-based anonymity is analyzed.

This section provides an accurate anonymity evaluation that goes beyond the unsuitable anonymity set size. Hence, this section is addressing **Research Question 11** as well.

IDENTIFYING LEAF SUBJECTS AS RECIPIENTS Following the communication model in Section 2.2.1, subjects v are only joining an communication overlay O_t if one of the following requirements is fulfilled:

1. v is a sender: $v = s \in \mathcal{S}_t$; v has to join the communication overlay O_t to be able to emit messages.

2. v is a recipient: $v = r \in \mathcal{R}_t$; v has to join the communication overlay O_t to be able to receive messages.
3. v is a broker, helping to connect a sender $s \in \mathcal{S}_t$ with recipients $r \in \mathcal{R}_t$: $v = b \in \mathcal{B}_t$ such that $d^+(v) \geqslant 1$

Assuming that the subject v is neither sender s nor recipient r, but is positioned as a leaf of the communication overlay O_t. As v is neither sender nor recipient, v has to be a broker b. A broker, however, joins the overlay only to relay messages from senders to recipients, thus, v must have at least one outgoing connection $d^+(v) \geqslant 1$. $d^+(v) \geqslant 1$ contradicts the statement that v is positioned in a leaf position and is neither sender nor recipient.

As a result of this, an global passive adversary $\mathfrak{A}_{p,g}$ can identify subjects in leaf position as recipient (or respectively as sender, which is considered in Chapter 6).

RECIPIENTS IN LEAF POSITION The path consolidation in ACO-based communication overlays also reduces the number of leaf subjects and thus increases the anonymity for recipients.

Figure 37 visualizes the number of recipients in leaf position $ls(O_t)$ for the three analyzed network types.

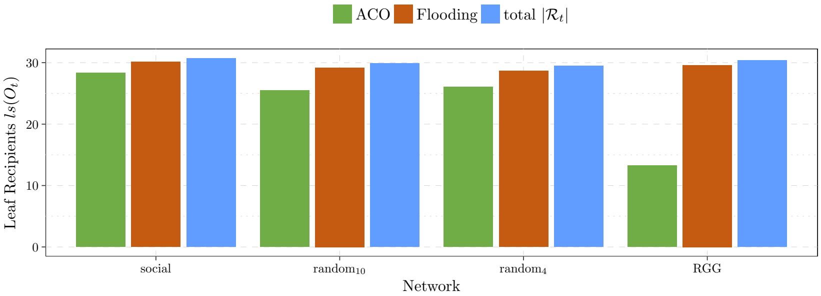


Figure 37: Number of recipients in leaf position $ls(O_t)$ of the established communication Overlays. The network types are arranged at the x-axis, the mean number of recipients in leaf position is placed at the y-axis—less recipients in leaf position $ls(O_t)$ is better.

The *small* but densely connected core on *social networks* prevents the path consolidation from ACO-based communication overlays from significantly reducing the number of recipients in leaf positions. The mean number of recipients in leaf positions $ls(O_t)$ on ACO-based communication overlays is 28.34, on flooding-based communication overlays 30.12.

The number of recipients in leaf position on ACO-based communication overlays and flooding-based communication overlay on *random networks* is similar to the number of recipients in leaf position on social networks.

The connection structure on *RGG* enables ACO-based communication overlays to reduce the number of recipients in leaf positions to 13.28 compared to 29.60 on flooding-based communication overlay.

Table 15 summarizes the number of recipients on leaf positions for both overlay establishment mechanisms as well as the respective ratio of recipients in leaf positions.

Table 15: Number of leaf recipients $\mathrm{ls}(O_t)$ on ACO-based and flooding-based communication overlays.

	ACO	Flooding	total
Social:			
$\mathrm{ls}(O_t)$	28.34	30.12	30.72
ratio	0.92	0.98	
Random$_{10}$:			
$\mathrm{ls}(O_t)$	25.47	29.20	29.87
ratio	0.85	0.98	
Random$_4$:			
$\mathrm{ls}(O_t)$	26.07	28.67	29.47
ratio	0.88	0.97	
RGG:			
$\mathrm{ls}(O_t)$	13.28	29.60	30.36
ratio	0.44	0.97	

improving leaf-mixture by topology adaptation

Daubert et al. [37] propose two mechanisms, *probabilistic forwarding* and *shell game*, to increase the number of ordinary subjects in leaf positions to lower the identification probability of recipients in leaf positions.

Using *probabilistic forwarding*, subjects in leaf positions may include adjacent neighbors in the communication overlay by extending the dissemination paths. The number of recipients in leaf positions is reduced proportional to the forwarding probability. However, by adding additional subjects to the communication overlay the efficiency is again impaired.

Using the *shell game*, subjects that join a communication overlay may swap positions with adjacent neighbors (that are already in the communication overlay). With that, the recipients can move to inner locations of the communication overlay. The swap is controlled with an exponential decay function, to reduce the number of swaps over time and to stop the shell game eventually. As the shell game does not involve additional subjects, the communication efficiency of the communication overlay stays stable.

Both probabilistic forwarding and the shell game relax the previously described requirement 3; it is no longer required that subjects in leaf positions are recipients. More details on the functionality of both mechanism are presented in Section 6.3.2.

Depending on the ratio of recipients in the system $\mathcal{R}_t/\mathcal{V}$, the shell game can be sufficient to decrease the ratio of recipients in leaf position below 0.5. For larger ratios, the communication overlays O_t have to be extended using the probabilistic forwarding to include additional subjects to increase the anonymity of recipients. The optimal leaf recipient ratio is given with the ratio of recipients in the respective communication overlay $\mathcal{R}_t/\mathcal{V}_t$; the desired maximal ratio of recipients in leaf position is 0.5 as this reduces the identification probability of an adversaries $\mathfrak{A}_{p,g}$ to a coin flip when they inspect a single leaf-positioned subject.

ACHIEVABLE ANONYMITY With the application of both mechanisms *probabilistic forwarding* and *shell game*, the ratio of recipients in leaf positions can be leveled down below 0.5 [40]. This analysis assumes, therefore, the *worst case* ratio of 0.5 for the ratio of recipients in leaf position and follows the steps outlined in Equations (21)–(25) of Section 4.4.2 to compute the anonymity degree of the communication overlays.

The execution of the anonymity protection measures shell game and probabilistic forwarding change the identification probabilities in Equation (21) and (22) with the recipient ratio of 0.5 in leaf position as follows:

$$p_{inner'} = \frac{|\mathcal{R}_t| - 0.5 \cdot ls(O_t)}{|\mathcal{V}_t| - ls(O_t)}$$

$$p_{leaf'} = \frac{|\{r \in R \colon d^+(r) = 0\}|}{ls(O_t)} \leqslant 0.5$$

$p_{inner'}$ is the probability of a subject that is not in leaf position being a recipient. For that, the number of recipients not in leaf positions $(|\mathcal{R}_t| - 0.5 \cdot \mathrm{ls}(O_t))$ is divided by the number of subjects in the overlay that are not in leaf positions $|\mathcal{V}_t| - \mathrm{ls}(O_t)$.

Analogously, $p_{leaf'}$ is the probability of a subject that is in leaf position being a recipient. For that, the number of recipients in leaf positions $|\{r \in R : d^+(r) = 0\}|$ is divided by the number of leaf subjects (that do not only comprise recipients after the execution of the shell game) $\mathrm{ls}(O_t)$.

The sum of all properties has to be 1.0 to ensure valid probabilities for the computation of the entropy. Therefore, both the probability of inner subjects $p_{inner'}$ and the probability of leaf subjects $p_{leaf'}$ need to be normalized:

$$p_{inner} = \frac{p_{inner'}}{|\mathcal{V}_t| - \mathrm{ls}(O_t)}$$

$$p_{leaf} = \frac{p_{leaf'}}{\mathrm{ls}(O_t)}$$

After the normalization, the computation of the anonymity degree ad follows the steps outlined by Equation (23)–(25) without changes:

1. establish the probability vector $\mathbf{p_k}$ from p_{inner} and p_{leaf}. (Equation (23))
2. compute the Shannon entropy S over the established probability vector $\mathbf{p_k}$. (Equation (24))
3. compute the anonymity degree ad by normalizing the Shannon entropy S with the maximal entropy S_{max}; $S_{max} = \log_2(|\mathcal{V}_t|)$ assumes that all subjects in the communication overlay $\mathcal{V}_t$ have the same identification probability. (Equation (25))

Social networks provide the following anonymity degree with their mean anonymity set size $ass = \mathcal{V}_t = 61.7$ and mean number of leaf nodes $\mathrm{ls}(O_t) = 28.34$. The identification probabilities of the subjects in the communication over are given by:

$$\begin{aligned} p_{inner'} &= \frac{|\mathcal{R}_t| - 0.5 \cdot \mathrm{ls}(O_t)}{|\mathcal{V}_t| - \mathrm{ls}(O_t)} \\ &= \frac{30.72 - 0.5 \cdot 28.34}{61.7 - 28.34} = 0.4961 \\ &\Rightarrow p_{inner} \approx 0.01487 \end{aligned}$$

$$\begin{aligned} p_{leaf'} &= 0.5 \\ &\Rightarrow p_{leaf} \approx 0.01764 \end{aligned}$$

These identification probabilities enable the computation of the Shannon entropy over the resulting probability distribution:

$$\begin{aligned} S &= -((|\mathcal{V}_t| - ls(O_t) \cdot p_{inner} \cdot \log_2(p_{inner}) \\ &\quad + ls(O_t) \cdot p_{leaf} \cdot \log_2(p_{leaf})) \\ &= -(33.36 \cdot 0.01487 \cdot \log_2(0.01487) \\ &\quad + 28.34 \cdot 0.01764 \cdot \log_2(0.01764)) \\ &= 5.924378 \end{aligned}$$

The normalized anonymity degree d is then computed by:

$$d = \frac{S}{S_{max}} = \frac{5.924378}{5.947199} = 0.9961628$$

Diaz et al. [44] suggest an anonymity degree $d \geqslant 0.8$ to be sufficient for anonymous communication. AnonPubSub [37] provides with the protection measures for recipients, *probabilistic forwarding* and *shell game*, with the novel ACO-based overlay establishment mechanism an anonymity degree of $d \approx 0.9962$ on social networks.

The anonymity degrees d for the *random*$_{10}$, *random*$_4$, and *RGG networks* are computed analogously:

RANDOM$_{10}$:

$$\begin{aligned} p_{inner'} = 0.3901 &\Rightarrow p_{inner} \approx 0.00888 \\ p_{leaf'} = 0.5 &\Rightarrow p_{leaf} \approx 0.01963 \\ d = 0.8981276 & \end{aligned}$$

RANDOM$_4$:

$$\begin{aligned} p_{inner'} = 0.2505 &\Rightarrow p_{inner} \approx 0.00382 \\ p_{leaf'} = 0.5 &\Rightarrow p_{leaf} \approx 0.01918 \\ d = 0.7462872 & \end{aligned}$$

RGG:

$$\begin{aligned} p_{inner'} = 0.3069 &\Rightarrow p_{inner} \approx 0.00621 \\ p_{leaf'} = 0.5 &\Rightarrow p_{leaf} \approx 0.03765 \\ d = 0.7730252 & \end{aligned}$$

Table 16 reports the anonymity degrees of ACO-based communication overlays as computed above as well as the anonymity

Table 16: Anonymity Degrees d on ACO-based and Flooding-based Communication Overlays.

	ACO	Flooding
Social	0.9962	0.9962
Random$_{10}$	0.8981	0.8322
Random$_4$	0.7463	0.7139
RGG	0.7730	0.7002

degrees of the respective flooding-based communication overlays.

The reason for the lower anonymity degree d on *random networks* and *RGGs* is the larger difference of the identification probabilities of inner subjects p_{inner} and p_{leaf}. With additional *shell game* position swaps (and, if required additional subjects included using the *probabilistic forwarding*), the anonymity can be improved.

The results of this analysis show that the novel ACO-based overlay establishment mechanism does not impair anonymity even though it improves the efficiency of the communication.

4.5 CONCLUSION

This chapter introduced a novel *ACO-based overlay establishment* mechanism. First, ACO-related background information is provided and it is argued why ACO is appropriate to be used in the establishment of communication overlays in anonymous group communication systems. Second, the ACO-based overlay establishment mechanism is introduced as a three-phase approach. Last, the novel ACO–based communication overlay is evaluated using a simulation study that ensures the usage of representative networks, evaluates the parameters of the ACO-based overlay establishment mechanism, and considers the influence on efficiency and anonymity.

The ACO-based overlay establishment mechanism is performed in a distributed fashion:

1. Recipients emit agents that perform a (biased) random walk and search senders. Agents bias their selection probabilities based on pheromones that have been placed by

preceding agents in phase 2, and based on their strictness factor. The strictness factor scales the importance of pheromones to foster either exploration of the neighborhood or exploitation of already found paths.

2. When agents have found a sender, they return to their recipient and mark the connections on their path with pheromones to bias the selection probabilities of subsequent agents in phase 1. Also, pheromones evaporate in this phase to enable further optimizations.
3. Recipients emit agents that activate the strongest known pheromone trail towards the senders regularly. This explicit activation enables the termination of the overlay establishment mechanism.

The key lessons of this chapter are as follows:

- ACO consolidates path and foster late message duplication by a combination of exploration and exploitation.
- The novel ACO-based overlay establishment mechanism creates communication overlays with less connections than the state of the art flooding-based communication overlays:
  - 5.69% less connections on *social networks*
  - 10.61% less connections on *random networks* with an average degree of 10 (random$_{10}$)
  - 6.71% less connections on *random networks* with an average degree of 4 (random$_{4}$)
  - 40.66% less connections on *RGGs*
- Due to the path consolidation, the communication delay increases:
  - 10.42%—or 0.5 hops—on *social networks*
  - 8.19%—or 0.37 hops—on *random networks* with an average degree of 10 (random$_{10}$)
  - 4.53%—or 0.27 hops—on *random networks* with an average degree of 4 (random$_{4}$)
  - 21%—or 1.8 hops—on *RGGs*
- Recipient protection measures like probabilistic forwarding and the shell game [37] remain intact as the ACO-based overlay establishment mechanism does only change the selection of connections for the communication overlays.

- The efficiency improvement is achieved while maintaining anonymity at the same level as it is achieved with flooding-based communication overlays. The anonymity degrees are:
  - 0.9962 on *social networks*
  - 0.8981 on *random networks* with an average degree of 10 (random_{10})
  - 0.7463 on *random networks* with an average degree of 4 (random_{4})
  - 0.7730 on *RGGs*

  The anonymity degrees on overlay that are established with the ACO-mechanism are at least as high as the anonymity degrees on traditional flooding-based overlays.

  Diaz et al. [44] suggest that an anonymity degree of $\geqslant$0.8 is *sufficient* to provide anonymous communication.

The explicit overlay activation may leak information about the active participants of an communication overlay to an adversary. The following chapter discusses different options to establish the routing information exchange with which the communication overlay activation can be performed. Also, the impact of dynamic subjects on the communication overlays is studied in the next chapter.

5 ROUTING WITH THE TENSION OF EFFICIENCY, ANONYMITY, AND CHURN

This chapter introduces how the ACO-based overlay establishment mechanism increases the reliability of the communication system when the communication system faces subject churn, i.e., it faces joining, leaving, and failing subjects. The ACO-based overlay establishment mechanism increases the ratio of connected recipients by utilizing pheromones that agents deposited during the initial overlay establishment. Additionally, this chapter discusses the exchange of routing information during the overlay establishment—and the overlay repairing under the influence of churn—under the angle of efficiency and anonymity. The second part of this chapter introduces and discusses four different mechanisms to share routing information, each of which reveals its advantages and disadvantages.

This chapter is structured as follows: after providing a motivation in Section 5.1, the different possibilities of subject churn are introduced in Section 5.2 and evaluated using a study. After that, Section 5.3 introduces and discusses different mechanisms to share routing information in anonymous communication systems.

5.1 MOTIVATION

Routing information exchange ensures that every subject in the communication system, especially the subjects in a specific communication overlay O_t, have the necessary topological knowledge to relay messages from senders $s \in \mathcal{S}_t$ to recipients $r \in \mathcal{R}_t$.

The necessity to share knowledge about the topology to establish routing information highlights the tension of efficiency and anonymity:

- Efficiency is improved when more topological information, i.e., information about the connection structure and interests of the subjects, is available. Additional information improves the quality ("optimality") of routing decisions concerning the chosen optimization criteria. For example, the overhead of message dissemination, measured in the number of messages, can be minimized when routing the messages along a Steiner tree. To compute the Steiner tree, however, the subjects need to be aware of the overall connection structure, as well as of the position of senders and recipients in the overlay.

- Anonymity, in contrast, is reduced when additional information is available. The availability of global structural information, as well as the locations of senders and recipients, enables an adversary to identify both sender and recipient quickly. In a similar direction, partial information already enables the adversary to focus their attention and attack power towards specific areas of the communication system—or a specific subset of subjects.

Routing information exchange needs to consider and address this tension to establish efficient and anonymous communication overlays O_t. While the previous chapter considered the service provider as adversary $\mathfrak{A}_{p,g}$, the information leakage of the exchange of routing information is of particular interest for a local adversary $\mathfrak{A}_{\_,c}$. This adversary $\mathfrak{A}_{\_,c}$ can learn information from details like the *size of the routing information* included in messages or stored at subjects. Efficiency considers here the size of the routing information that is either transmitted in the messages or is stored by the subjects.

This tension is of particular relevance when churn, i.e., subjects joining and leaving the communication system, changes the topology from time to time. These topology changes enforce recipients to re-enter their overlay establishment state and adapt their connection to the communication overlay.

In the following sections, first, the impact of churn on the communication reliability is analyzed, and it is discussed how the ACO-based overlay establishment mechanism supports recipients to remain connected or, if necessary, enables them to reconnect fast. Second, several different several different mechanisms to share routing information are discussed and compared concerning their impact on efficiency and anonymity.

5.2 EFFICIENCY AND ANONYMITY IN THE PRESENCE OF CHURN

Centralized or decentralized systems can easily control the impact of churn as the service is provided by dedicated subjects that share their knowledge. However, churn is a challenge in distributed (P2P) systems where all subjects cooperate to provide the service.

In a distributed communication system, which is the focus of this thesis, churn changes the topology from time to time. These changes require the subjects $v \in \mathcal{V}$ to continuously adapt the structure of the communication overlay O_t to reconnect to the overlay when separated through leaving subjects. Similarly, newly joining recipients have to able to connect to the communication overlay, thereby altering the structure of the overlay O_t.

The following sections introduce the possible manifestations of churn and outline the implications that result from the churn. This introduction of churn is complemented with a discussion of properties of the novel ACO-based overlay establishment mechanism (see Chapter 4) that reduce the influence of subject churn. A simulation study shows then the impact of subject churn and the ability of ACO-based communication overlays to cope with it.

5.2.1 *Concept of Dynamic Subject Behavior*

Subjects are showing dynamic behavior for many reasons, for example, due to diurnal behavior or turning devices off overnight. As a result of this, two simplified classes of devices can be derived:

- *stable* devices that are (nearly) never leaving the system like servers which are always online except when, for example, shut down for maintenance—due to their reliability, they establish the simplified class of stable subjects. They may, however, join late such that they are not available the whole time.
- *dynamic* devices that may leave and join the system at any point in time. Exemplary devices in this class are smartphones, laptops, and PCs.

In research of *human-driven* P2P systems, for example, for the purpose of file-sharing, researchers have shown that churn follows exponential or Weibull distributions [66, 137, 151, 152]. The key lesson of this characterization is: the longer a subject is participating in the system, the higher the probability that the subject will participate at a next point in time considered.

A communication system needs to consider three classes of systems when handling subject churn:

- *joining* subjects: A subject can join the system and connect to other subjects. The subject itself can be completely new or re-join after leaving the system before. Depending on the interest of this subject, they may only contribute as a broker $b \in \mathcal{B}_t$ or they may take an active role as sender $s \in \mathcal{S}_t$ or recipient $r \in \mathcal{R}_t$.

- *leaving* subjects: A subjects can leave the system *after announcing* their intent to leave to all adjacent neighbors. These neighbors can then react and minimize the damage of the leaving subject.

- *failing* subjects: A subject can leave the system *without announcing* their interest to leave, for example, when the device crashes or loses its connection. This failure has to be detected by the neighbors on their own, for example, by detecting missing synchronization or heartbeat messages. Here, subjects have to repair already occurred damage in the communication overlay O_t.

An anonymous communication system that bases its anonymization and service provisioning on the P2P-principle needs to consider all three manifestations of subject churn by addressing the respective challenges.

5.2.2 *Addressing the Three Cases of Churn*

The ACO-based overlay establishment mechanism utilizes the pheromones (see Sections 4.3.3–4.3.5) to reduce and cope with the influence of all three cases of churn.

5.2.2.1 *Joining Subjects*

The set of joining subjects comprises both late-joining subjects and re-joining subjects.

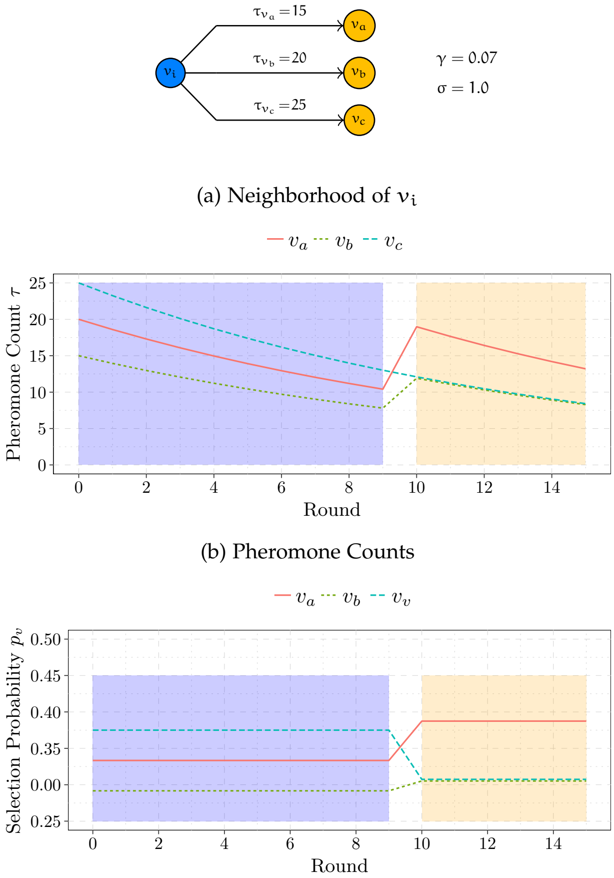
Recipients and senders that join an existing communication system are starting *their* instance of the overlay establishment mechanism in the phase 1, the discovery phase. The subject starts to emit agents in order to find either senders $s \in \mathcal{S}_t$ or recipients $r \in \mathcal{R}_t$ that are connected to the communication overlay O_t. During that search, agents recognize previously established pheromones on the connections and bias their random walk accordingly. As pheromones evaporate proportionally (see Equation (13)), the selection probabilities p_{v_i} remin stable. As a result of this, the stronger pheromone trails will still lead towards the senders or, in general, towards the communication overlay.

The utilization of pheromones that are placed in prior overlay establishment enables subjects to quickly (re-)connect to the communication overlay. The then newly placed pheromone markers dominate the pheromone-dependent portion of selection probability (see Equation (10)) quickly if the pheromone trails are "old". If the pheromone trails are fresh, they are strong enough to lead the joining subject towards the communication overlay. Example 22 exemplifies the influence of old and new pheromones.

Example 22: Figure 38b shows the pheromones on connections to three neighboring subjects v_a–v_c where pheromones evaporate with $\gamma = 0.07$. In round $rd = 10$, the connection to neighbor v_a then receives 10 additional pheromones, the connection to v_b receives 5 additional pheromones. While the relative distance stays stable with the proportional evaporation, and, as result of this, stable selection probabilities, the selection probabilities shift immediately with the new pheromones deposited; Figure 38c visualized the progression of selection probabilities.

rd	τ_{v_a}	τ_{v_b}	τ_{v_c}	(p_{v_a})	(p_{v_b})	(p_{v_c})
0	20.00	15.00	25.00	0.3333	0.2917	0.3750
1	18.60	13.95	23.25	0.3333	0.2917	0.3750
…	…	…	…	…	…	…
9	10.41	7.81	13.01	0.3333	0.2917	0.3750
10	18.98	11.91	12.10	0.3874	0.3052	0.3074
…	…	…	…	…	…	…
15	13.20	8.29	8.42	0.3874	0.3052	0.3074

Proposition 6 *The ACO-based overlay establishment mechanism can utilize old pheromones and combine them with newly deposited pheromones. As a result of this, the agents may exploit previously established knowledge, while the evaporation ensures that agents are not locked in old paths.*



(a) Neighborhood of v_i

(b) Pheromone Counts

(c) Selection Probabilities

Figure 38: In the first phase (blue) on the left hand side, pheromones evaporate proportional according to Equation (13), the selection probabilities remain stable. In round 10, the connection to v_a receives 10 pheromones, the connection to v_b receives 5 pheromones; the selection probabilities p_v change accordingly.

5.2.2.2 *Leaving Subjects*

Leaving subjects announce their intent to leave the communication system. This announcement is intended to enable the remaining subjects to react and rebuild the communication overlay without the leaving subject.

For that, the leaving subject v_k sends a message m^{leave} to all adjacent subjects $v_l \in N^{\text{-}}(v_k)$. These subjects perform the following two tasks:

- *mark* v_k *as inactive*: prevent the activation of overlay-paths that rely on v_k

- *inform affected neighbors*: evaluate whether v_k is part of the communication overlay O_t by checking whether v_k is part of own routing tables. Forward leave notification along overlay paths to enable recipients $r \in \mathcal{R}_t$ to react and to start emitting agents

Recipients that receive a leave notification will enter the first phase of the ACO-based overlay establishment mechanism and start emitting agents. As Section 5.2.2.1 describes and Example 22 exemplifies, the emitted agents utilize previously deposited pheromones to detect paths around the leaving subject.

5.2.2.3 *Failing Subjects*

In contrast to leaving subjects, failing subjects are not announcing their leave beforehand, for example, as a result of a system crash. This is an extreme case of the previous churn type of *leaving subjects*.

As failing subjects v_k do not announce their leaving, all subjects have to check the liveness of their neighbors using regular *heartbeat* messages. When a subject v_k fails to respond to these messages, the neighbors realize that the subject failed and will perform similar tasks to the case of leaving subjects:

- *mark* v_k *as inactive*: prevent the activation of overlay-paths that rely on v_k

- *inform affected neighbors*: evaluate whether v_k is part of the communication overlay O_t by checking whether v_k is part of own routing tables. Forward fail notification along overlay paths to enable recipients $r \in \mathcal{R}_t$ to react and to start emitting agents

Recipients that receive a leave notification will enter the first phase of the ACO-based overlay establishment mechanism and start emitting agents. As Section 5.2.2.1 describes and Example 22 exemplifies, the emitted agents utilize previously deposited pheromones to detect paths around the leaving subject.

Proposition 7 *Agents of joining subjects will speedup their discovery with pheromones that are deposited in early overlay establishment phases. The usage of these old pheromones increases the bootstrapping speed while new pheromones that dominate the selection probability will ensure the liveness of found paths.*

Proposition 8 *Utilization of previously deposited pheromones help to counter disruptions of communication overlays through leaving and failing subjects. The communication will, therefore, be more robust against churn.*

5.2.3 *Evaluation of Churn-resistant, Efficient, and Anonymous Communication*

This section analyzes the effect of subject churn on anonymous communication overlays and the effectiveness of the proposed reactions using the ACO-based overlay establishment mechanisms.

Using the simulation framework from Chapter 4, a simulation study is performed and described in Section 5.2.3.1; the evaluation metrics are described in Section 5.2.3.2. Section 5.2.3.3 presents and discusses the results.

5.2.3.1 *Simulation Study*

The simulation is performed on the very same networks as the original overlay establishment mechanism. Therefore and because of the results of the network configuration simulation in Section 4.4.3, the churn simulation study is performed on:

- *Social Networks* following the Barabàsi-Albert model with $m = 3$. Thus, every subject added to the network adds three connections.

- *Random Networks* following the Erdős-Rényi model with $\overline{d(v)} = 4$. Thus, every subject in the network has on average four connections.

- *Random Geometric Graphs RGG* with random positioning of the subjects on a euclidean plane of dimension $dim = 2$, and the connectivity threshold $ct = 0.1$.

All networks are populated by $2,000$ subjects.

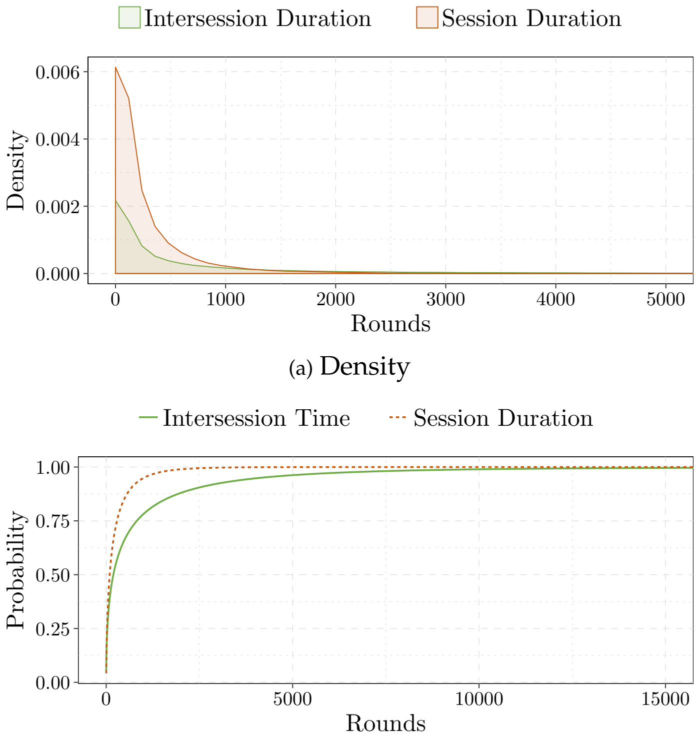
A ratio of 0.015 of the subjects is randomly selected to be recipients of the established communication overlay.

In the ACO configuration, recipients are emitting agents in Phase 1 (“*Sender Discovery*”) for $phase1_{dur} = \{200, 500, 1,000\}$ rounds. Joining recipients and recipients that react to leave and fail notifications emit agents for the same time.

The session duration (lifetime) and intersession time are configured following the results of [66, 137] using two Weibull distributions:

- Intersession time: shape $k_{it}=0.61511$, scale $\lambda_{it}=413.6765$
- Session duration: shape $k_{sd}=0.47648$, scale $\lambda_{sd}=169.5385$

Figure 39 visualizes the respective density functions and cumulative distribution functions. Both Stutzbach et al. [137] and Gross et al. [66] suggest that the intersession times are significantly smaller than the session duration; as a result of this, it is expected that more subjects leave the system than join the system.



(a) Density

(b) Cumulative Distribution Function

Figure 39: Weibull distributions for intersession times and session durations, parameterized following [66].
For purpose of visualization, the x-axes are limited to 5,000 and 15,000 rounds respectively.

At the beginning of the experiment, the communication system establishes an communication overlays without being affected by churn. The evaluation of the ACO-mechanism in Sections 4.4.4 and 4.4.5 suggests that the recipients complete the overlay establishment in less than 500 rounds. As a result of this, churn is activated in round 500, i.e., all subjects draw a round from the session duration distribution and initiate a leave or fail event as soon as their respective round is due. As soon as their leave or fail event is initiated, the subjects draw another round from the intersession time distribution to schedule their join-event. After additional 1,000 rounds, in which subjects may leave the system, churn is restricted again to allow only join-events, i.e., after round 1,500, only join-events are executed by the subjects, and leave-events and fail-events are discarded. The experiment continues then until round 10,000 to provide the subjects with enough time to stabilize again, i.e., enough time to reconnect to the communication overlay.

Table 17 summarizes the configuration of the simulation study.

Table 17: Simulation Parameters

Parameter	Value
Runs	25
Rounds per run	20,000
Network size ($\|\mathcal{V}\|$)	2,000 (1,000 on RGG)
Recipients ($\|\mathcal{R}_t\|$)	$0.015 \cdot \mathcal{V} \Rightarrow \|\mathcal{R}_t\| = 30$
Network types	{social; random$_4$; RGG}
Connectivity social network	$m_0 = 2$, $m = 3$
Average degree random network	$\overline{d(v)} = 4$
Connectivity threshold RGG	0.1
ACO Configuration:	see Table 8 in Section 4.4.4 (p. 97)
Evaporation γ	0.07 | 0.01
Strictness σ	[0.75, 1.0] | [0.5, 1.0]
Agents per Recipient per Round k	20
Phase 1 Duration $phase1_{dur}$	{200; 500; 1,000 }
Churn Configuration:	
Distribution	weibull
Intersession time	$k_{it} = 0.61511$, $\lambda_{it} = 413.6765$
Session duration	$k_{sd} = 0.47648$, $\lambda_{sd} = 169.5385$
Churn start	500
Churn end	1,500

5.2.3.2 *Evaluation Metrics*

With the performed simulation study the following metrics are computed to evaluate the effect of subject churn on the communication system, the definition of the metrics itself follows the simulation study in Section 4.4:

- *Recipient Success Ratio* $\mathrm{sr}_{\mathcal{R}_t}(O_t)$: Computes the ratio of connected recipients and, thus, measures the robustness against subject churn.
- *Weight* $w_{\mathcal{E}}(O_t)$: Computation of the communication overlay size, measured by counting the number of connections.
- *Communication Delay* $\mathrm{cd}_{avg}(O_t)$: Computation of the distance between sender and recipient, averaged over all sender recipient pairs.
- *Leaf Subjects* $\mathrm{ls}(O_t)$: Computes the number of recipients in leaf positions. The number of leaf subjects can also be used as indicator for the instability of the communication overlay: an unstable number of leaf subjects indicates frequent path disruptions and the resulting repair actions.

5.2.3.3 *Communication under Churn*

This section presents and discusses the results of the robustness of the communication system against subject churn. The following challenges and research questions are answered:

- Is duration in which recipients emit agents improving the robustness against subject churn?
- To which extent are subjects able to receive messages while the communication system is affected by subject churn?
- Is churn increasing the communication costs for the remaining subjects?
- How is churn affecting the communication delay?

THE INFLUENCE OF SUBJECT CHURN Figure 40 visualizes the number of inactive subjects over time; the blue highlighted area is the time in which subjects may leave the system. It shows the massive effect of subject churn on the availability of subjects in an anonymous communication system by visualizing the number of *inactive* subjects, i.e., visualizes the number of subjects that left or failed. The differing churn distributions cause the quick rise in the number of inactive subjects while these subjects are only slowly re-joining the system. About 400 subjects turn inactive in the 1,000 rounds between 500 and 1,500 in which the leave events and fail events are performed while only about 40 of these subjects rejoin in the 3,500 rounds until round 5,000.

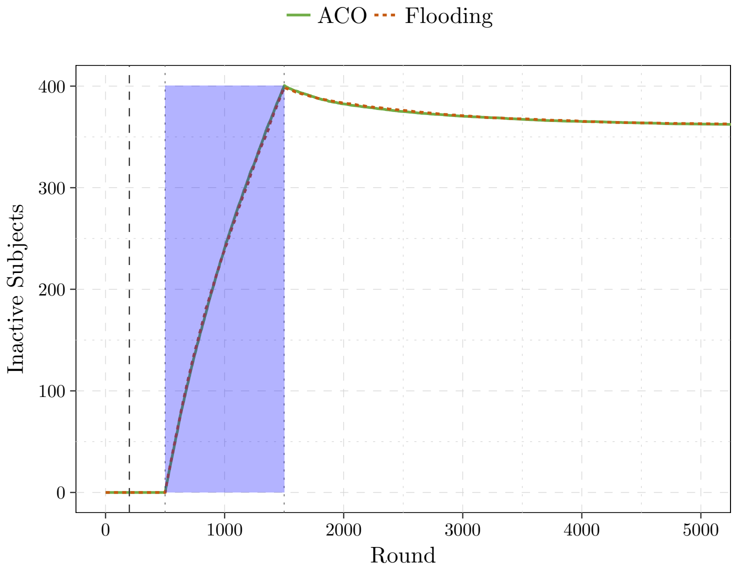


Figure 40: Impact of churn on social networks: Measuring the number of leaving, failing, and joining subjects. Within the blue highlighted interval, subjects may leave the system; the dashed line marks the end of the initial overlay establishment.

RECIPIENT SUCCESS RATIO As described above, churn has a massive affect on the availability of subjects. Continuously leaving subjects disrupt the connection overlay O_t frequently. Figure 41 shows that both overlay establishment mechanisms are hardly able to cope with frequent leave-events.

Notably, the novel ACO-based overlays achieve a success ratio of 0.89–0.90 on social networks, a success ratio of 0.89–0.87 on random networks, and a success ratio of 0.87–0.89 on RGGs. These high success ratios are facilitated by the ability to react fast to a changing environment due to the pheromone placement.

In contrast to that, flooding-based communication overlays achieve only a success ratio of 0.60–0.65 on social networks, a success ratio of 0.54–0.59 on random networks, and a success

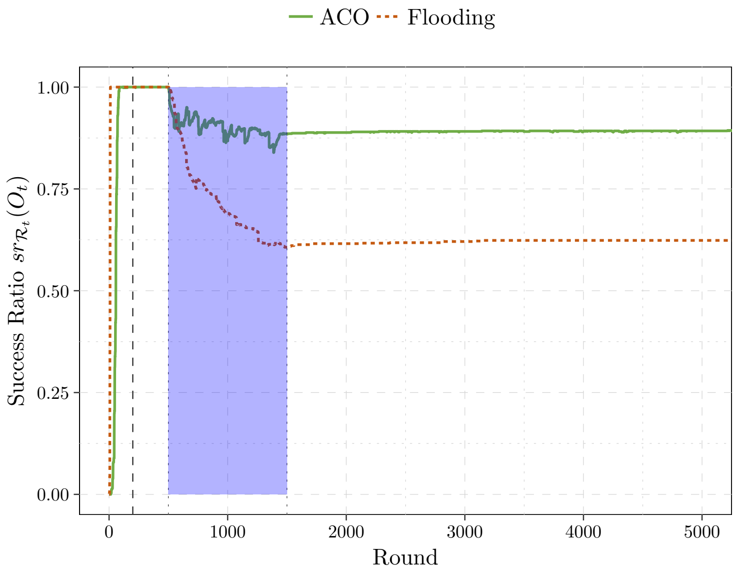


Figure 41: Success ratio $sr_{\mathcal{R}_t}(O_t)$ of overlays under churn over time. Within the blue highlighted interval, subjects may leave the system; the dashed line marks the end of the initial overlay establishment.

ratio of 0.48–0.52 on RGGs. Even when being able to be reconnected, recipients have to perform their search for the communication overlay O_t, the sender s, or recipients r every time from scratch without utilization of previously established knowledge.

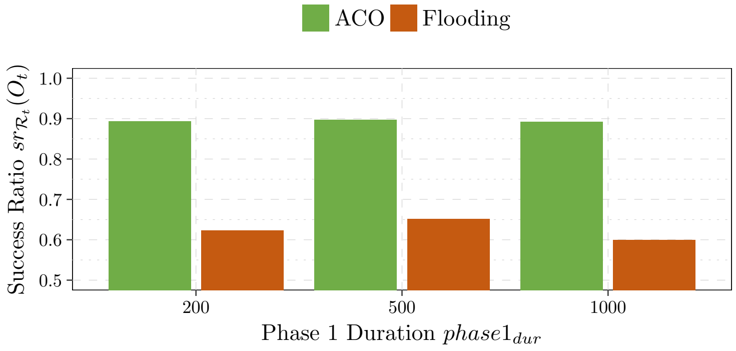


Figure 42: Success ratio $sr_{\mathcal{R}_t}(O_t)$ of overlays after 10,000 rounds under churn. *As the ratio is constantly over 0.5, the y-axis is limited to [0.5, 1.0] to highlight the relative differences.*

Figure 42 visualizes the success ratios for varied durations of Phase 1 on social networks. Table 18 reports the respective success ratios on all inspected networks. Both show that the success ratios of the ACO-based communication overlays are independent from the duration $\mathtt{phase1}_{\mathtt{dur}}$ for which recipients emit agents and consistently significantly higher than on conventional overlays.

Table 18: Success Ratio under churn after 10,000 rounds.

Network	$\mathtt{phase1}_{\mathtt{dur}}$	ACO	Flooding
social	200	0.8936	0.6236
	500	0.8976	0.6520
	1,000	0.8920	0.5996
random_4	200	0.8688	0.5768
	500	0.8254	0.5880
	1,000	0.7979	0.5428
RGG	200	0.8778	0.5184
	500	0.8850	0.5008
	1,000	0.8743	0.4800

COMMUNICATION OVERLAY WEIGHT Leaving subjects may cause disruptions in a communication overlay; to keep up the success ratio and to keep the recipients connected to their communication overlay, the ACO-based overlay establishment mechanism has to establish a path around the disruption. The overlay weight, i.e., the number of connections in the communication overlay, is therefore expected to increase. Figure 43 visualizes the overlay weight on social networks over the time. It shows that the overlay weight is only slightly increasing. The densely connected core of the social network enables the ACO mechanism only moderately to increase the overlay weight. For that, the agents can explore the dense core to establish "nearby" paths around the disruption which avoid a steep increase of the overlay weight.

Neither random networks nor RGGs provide such a densely connected core. While the ACO mechanism establishes still overlays with comparable success ratios, the overlay weight is increasing more severely.

The overlay weight on conventional flooding-based communication overlays is decreasing as churn removes parts of the overlay where the recipients cannot not recover from as reported in the analysis of the success ratio before.

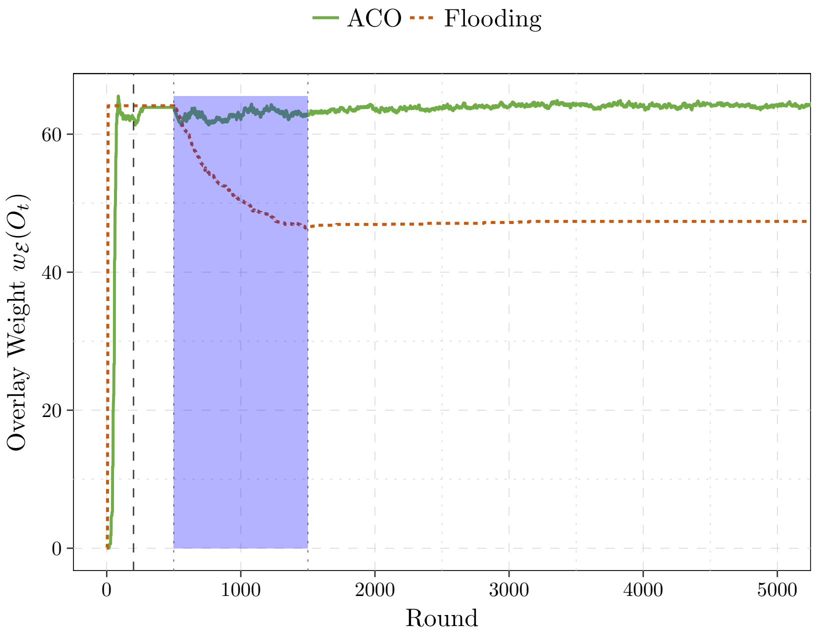


Figure 43: Weight $w_\mathcal{E}(O_t)$ of overlays under churn over time. Within the blue highlighted interval, subjects may leave the system; the dashed line marks the end of the initial overlay establishment.

Table 19 summarizes the weights $w_\mathcal{E}(O_t)$ of the established overlays; for the comparison, the overlay weights of the corresponding networks without influence of churn (see Section 4.4.5) are included. Also, the weights of the flooding-based overlays are included for completeness.

Table 19: Overlay Weight $w_\mathcal{E}(O_t)$ under Churn after 10,000 rounds. *Flooding-based overlays are included for completeness even though they miss to connect 40%–50% recipients.*

Network	ACO	without Churn (Section 4.4.5)	*Flooding*
social	64.08	62.2 (+3.04%)	*47.3600*
random$_4$	107.44	90.93 (+18.16%)	*73.3200*
RGG	105.39	59.83 (+76.15%)	*70.5600*

COMMUNICATION DELAY Churn causes the overlay weight $w_\mathcal{E}(O_t)$ to increase; the increasing number of connections can result in an increasing communication delay $cd_{avg}(O_t)$ which is caused by longer path lengths between sender and recipients in a communication overlay.

Figure 44 visualizes the average path length between senders and recipients in an overlay under churn over time. As such, it depicts that the average path length between sender and recipients is unstable under churn, i.e., the paths between sender and recipients change frequently. The effect is similar to the overlay weight $w_{\mathcal{R}_t}(O_t)$: on social networks, the communication delay $\mathrm{cd}_{\mathrm{avg}}(O_t)$ remains of stable order (even though shaky over time). On random networks and on RGGs, communication delay $\mathrm{cd}_{\mathrm{avg}}(O_t)$ increases by 2–4 hops.

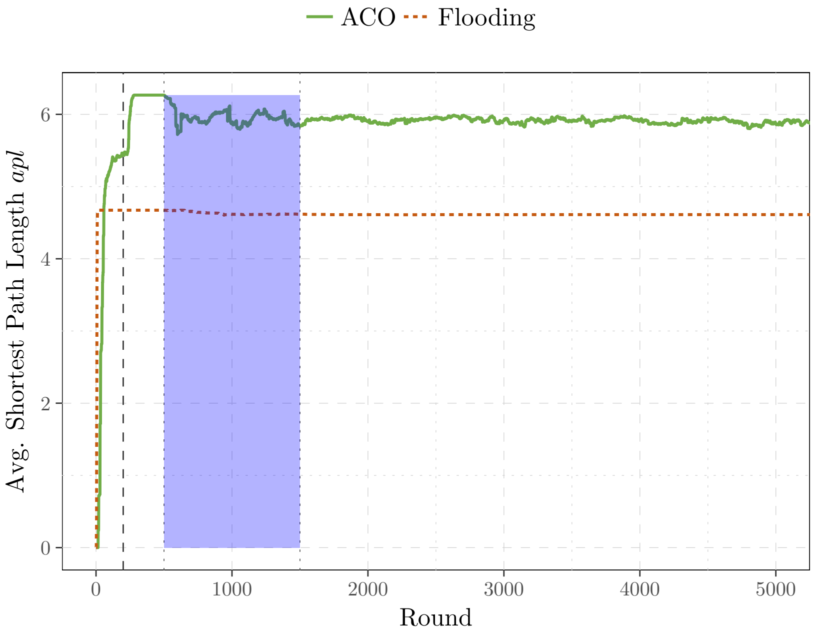


Figure 44: Average path length apl between the sender and a recipient. Within the blue highlighted interval, subjects may leave the system; the dashed line marks the end of the initial overlay establishment.

Table 20 summarizes the communication delay $\mathrm{cd}_{\mathrm{avg}}(O_t)$ under churn after 10,000 rounds. for the comparison, the communication delay $\mathrm{cd}_{\mathrm{avg}}(O_t)$ of the corresponding networks without influence of churn (see Section 4.4.6) are included. Also, the communication delay $\mathrm{cd}_{\mathrm{avg}}(O_t)$ of the flooding-based overlays are included for completeness.

LEAF SUBJECTS Under churn, subjects have to react to leaving subjects by establishing alternative paths between senders and

Table 20: Communication Delay $cd_{avg}(O_t)$ under Churn after 10,000 rounds. *Flooding-based overlays are included for completeness even though they miss to connect 40%–50% recipients.*

Network	ACO	without Churn (Section 4.4.5)	*Flooding*
social	5.92	5.3 (+11.70%)	*4.61*
random_4	8.23	6.23 (+31.10%)	*5.85*
RGG	12.25	8.6 (+42.44%)	*6.50*

recipients. As the ACO-based overlay establishment mechanism aggregates paths, it is likely that the number of leaf subjects $ls(O_t)$ changes when new paths are formed.

As such, the number of leaf subjects $ls(O_t)$ under churn is an indicator for the instability of the communication. Figure 45 visualizes the number of leaf subjects $ls(O_t)$ on social networks under churn over the time. The drop of the number of leaf subjects when churn is activated is caused by the many subjects that leave the system; during the time interval in which subjects may leave the system, the number of leaf subjects is shaky but stable in its order. When the overlay stabilizes as soon as there are no more disruptions, the number of leaf subjects stabilizes as well.

The increasing difference between the number of leaf subjects on ACO-based overlays and the number of leaf subjects on flooding-based overlays can be explained by the different behavior of the overlay establishment mechanisms. The ACO mechanism utilizes pheromones from previous overlay establishment phases to find the sender quickly. As a result of that, the paths are likely to be shared. As only churn-affected recipient return into the overlay establishment phase, only these recipients emit agents—therefore, only these can connect themselves to other recipients which will only occur if they directly "find" another recipient. As a result of this, the number of leaf subjects remains reasonably high. The steep drop of the number of leaf subjects on flooding-based overlay is caused by the massive number of recipients that remain unconnected from the communication overlay.

Table 21 summarizes the number of leaf subjects under churn after 10,000 rounds. for the comparison, the number of leaf subjects of the corresponding networks without influence of churn (see Section 4.4.6) are included. Also, the number of leaf subjects of the flooding-based overlays are included for completeness.

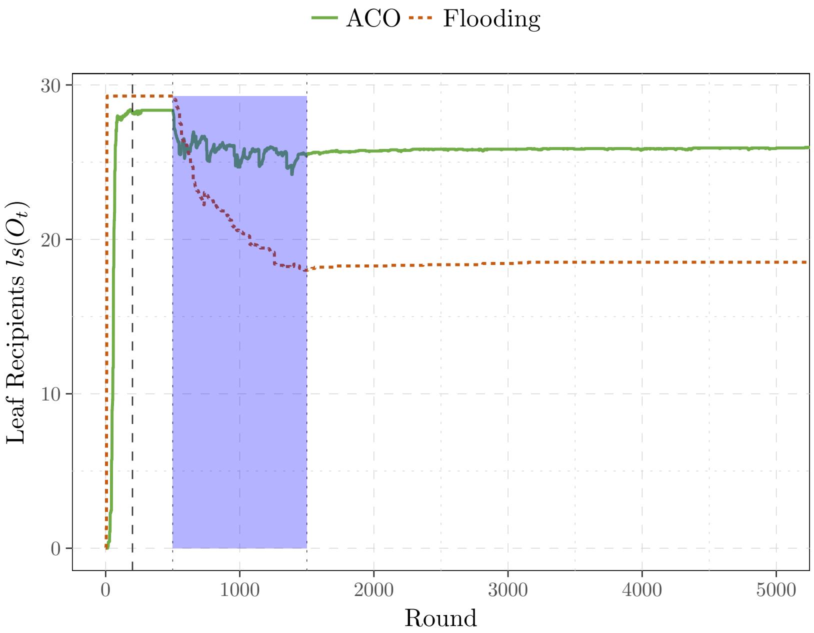


Figure 45: Number of recipients in leaf positions $\mathrm{ls}(O_t)$ in a churn affected communication overlay. Within the blue highlighted interval, subjects may leave the system; the dashed line marks the end of the initial overlay establishment.

Table 21: Number of leaf subjects $\mathrm{ls}(O_t)$ under Churn after 10,000 rounds. *Flooding-based overlays are included for completeness even though they miss to connect 40%–50% recipients.*

Network	ACO	without Churn (Section 4.4.5)	*Flooding*
social	25.96	28.34 (-8.40%)	*18.52*
random_4	24.48	26.07 (-6.10%)	*16.88*
RGG	16.22	13.28 (-22.14%)	*14.92*

SUMMARY ACO-based communication overlays improve the recipient success ratio over state of the art flooding-based overlays and provide a functional communication overlay O_t for about 90% the recipients. Even when facing immense subject churn, the communication delay and communication overhead remain in the same order of magnitude.

However, frequent leave and fail events require a continuous exchange of routing information. The next section will discuss how to share this routing information in the tension of efficiency and anonymity.

5.3 ROUTING WITH CONSIDERATION OF EFFICIENCY AND ANONYMITY

Routing of messages between senders $s \in \mathcal{S}_t$ and recipients $r \in \mathcal{R}_t$ requires information about the overlay structure O_t. With the inherent need of information to achieve successful and efficient transmission of messages, routing is obviously in tension between the conflicting goals of efficiency and anonymity. Simply put, the more available information, the higher the efficiency; the less information available, the better the anonymity.

In this section, first, the concept of routing is discussed in the light of this tension of efficiency and anonymity. After that, four possible approaches to routing are introduced, followed by a qualitative evaluation of the routing information exchange mechanisms.

5.3.1 *Concept of Routing in the Tension*

Routing enables the communication system to transmit messages between senders $s \in \mathcal{S}_t$ and recipients $r \in \mathcal{R}_t$. All subjects share routing information during the overlay establishment. After overlay establishment, this information is memorized and either available at intermediate subjects that have to relay messages or included in the messages itself.

However, the amount of information available for routing enables a trade-off between efficiency and anonymity. More information enables efficient routing, i.e., provides optimal decisions for the forwarding of messages by subjects, at the cost of providing more information to an adversary $\mathfrak{A}$. Routing information encompasses structural properties of the communication system, i.e., information about senders, recipients, and brokers. Using this information, an adversary $\mathfrak{A}$ can reduce the anonymity sets by excluding subjects from the anonymity set and collecting evidence that increases their probability of identifying senders or recipients. Reducing information that is used—and shared—for routing limits the evidence that an adversary $\mathfrak{A}$ can learn; the system improves the anonymity protection. However, by reducing the routing information, subjects are no longer able to make optimal—or near optimal—decisions as necessary information is missing.

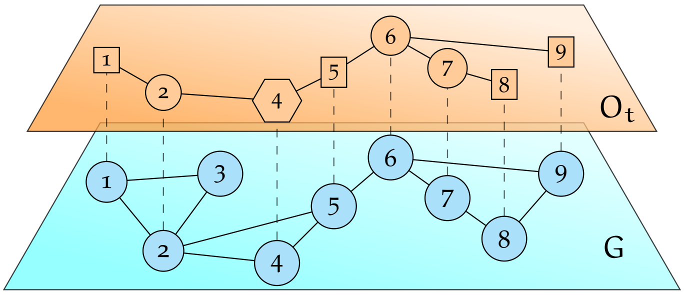
Naturally, by distributing functionality and responsibility to the subjects in a P2P manner, the available knowledge of each participant is restricted to a local view—if the information is not further exchanged and accumulated. Routing in the tension of efficiency and anonymity aims at providing enough information to achieve routing with reasonable efficiency while still preventing an adversary $\mathfrak{A}$ from gaining additional knowledge.

5.3.2 *Adversary Model for Analyzing Information Leaks of Routing Information Exchange Mechanisms*

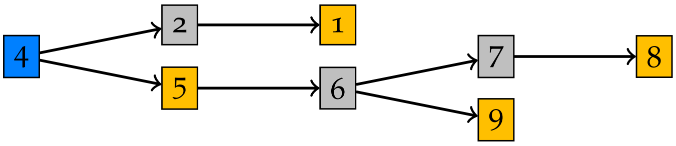
In the previous Chapter 4 and the previous Section 5.2, a global adversary $\mathfrak{A}_{p,g}$ (see Section 2.3) was used. This $\mathfrak{A}_{p,g}$ adversary resembles a strong adversary, for example, the service provider. In this analysis of mechanism for exchanging routing information, a different adversary instantiation is appropriate. The global adversary $\mathfrak{A}_{p,g}$ is not depending on collecting evidence from routing information; they can trace messages directly on the network and obtain the same—or even more—evidence directly. However, a locally restricted adversary $\mathfrak{A}_{\_,l}$ (see Section 2.3) is not able to trace messages from this global perspective, yet, they may be able to collect evidence from leaked information from the routing information. Thus, the evaluation of mechanism for routing information exchange focusses on the analysis of a limited adversary $\mathfrak{A}_{\_,l}$.

5.3.3 *Four Methods to Exchange Routing Information*

Routing information is usually distributed using one of two mechanisms. First, routing information can be *included in the message* itself. Every subject checks the respective information to decide the to which neighbor the message is forwarded. While this mechanism is inherently simple, adversaries $\mathfrak{A}$ can also obtain the routing information if they handle a message. Second, routing information can be *maintained by the subjects* themselves by establishing local routing tables. The messages itself are then only equipped with a topic and a message-id to enable subjects checking their local routing tables and determine the next hop for the message.



(a) Topology based on Figure 8.



(b) Dissemination Tree.

Figure 46: Communication system, using the exemplary scenario depicted in Figure 8: One sender (4; blue), four recipients (1, 5, 8, and 9; orange), and three brokers (2, 6, 8)

Research Question 12 *How can routing be established with the conflicting optimization goals efficiency and anonymity? What is the tradeoff between both and can it be controlled?*

Based on the topology used in Figure 8 in Section 2.2, the evaluation in this section is using the topology shown in Figure 46. Derived from this topology, the required routing information for the overlay O_t is visualized in Figure 46b. In this example, subject 4 is the sender, subjects 1, 5, 8, and 9 are recipients, and subjects 2, 6, and 7 are brokers.

The following sections introduce four variants of routing information exchange, namely successor lists, successor lists with multi-layered encryption, Bloom filter-contained routing information, and distributed routing tables.

5.3.3.1 *Successor Lists*

Using successor lists, routing information is included in communication messages as an encrypted list of subject-ids. The encryption is based on a system-wide shared key to prevent subjects outside of the system from obtaining the routing information.

When forwarding a message using this technique, subjects check the routing information, i.e., check the successor list, that is embedded in the message for the next hop(s), i.e., for their successor(s) in the communication overlay. The successor list that is included in the messages is being collected during the overlay establishment, for instance, in Phase 3 of the ACO-based overlay establishment (see Section 4.3.5).

In the exemplary scenario given in Figure 46, the message is sent from sender $s = 4$ to the recipients $r \in \mathcal{R}_t = \{1, 5, 8, 9\}$; thus, the list of routing information looks like the following list, where junctions are indicated by curly braces:

$$\mathtt{path} = (\{(2, 1), (5, 6, \{(7, 8), (9)\})\})$$

Each subject strips unnecessary information from this successor list, for example, their own id and the alternative path in case of junctions. In the example, subject 4 would duplicate the message and include $(2, 1)$ and $(5, 6, \{(7, 8), (9)\})$ respectively before forwarding the message to subjects 2 and 5.

> **Example 23:** The routing information included for the succeeding transmission steps in the given example scenario (see Figure 46) are as follows:
>
> 1. round
> $4 \rightarrow 2$: $\mathtt{path} = (2, 1)$
> $4 \rightarrow 5$: $\mathtt{path} = (5, 6, \{(7, 8), (9)\})$
> 2. round
> $2 \rightarrow 1$: $\mathtt{path} = (1)$
> $5 \rightarrow 6$: $\mathtt{path} = (6, \{(7, 8), (9)\})$
> 3. round
> $6 \rightarrow 7$: $\mathtt{path} = (7, 8)$
> $6 \rightarrow 9$: $\mathtt{path} = (9)$
> 4. round
> $7 \rightarrow 8$: $\mathtt{path} = (8)$

Proposition 9 *Successor lists disseminate topological information using plain lists of subject-ids. They minimize overhead by avoiding additional protection measures and ensure routing in the limits of the overlay establishment mechanism.*

EFFICIENCY Using successor lists, senders include all routing information in the messages. For that, senders save the lists with routing information and have, thus, to provide enough memory for the lists. The necessary memory equals the size the addresses $|v.\mathtt{address}|$ that are used for the subjects, for example, 128 bit for an IPv6 address, times the number of subjects of which the addresses have to be included:

$$\mathtt{s.mem} = |\mathcal{V}_t| \cdot |\mathtt{v.address}| - |\mathtt{v.address}| \tag{26}$$

The communication overhead is defined by the path length succeeding each subject; therefore, the upper limit of the communication overhead is equal to $\mathtt{s.memory}$. The *average communication overhead* is defined by the apl of the communication overlay $\mathtt{apl}(O_t)$ times the address size:

$$\mathtt{com}_{oh} = \mathtt{apl}(O_t) \cdot |\mathtt{v.address}| \tag{27}$$

ANONYMITY As the successor list contains clear routing information, the adversary $\mathfrak{A}\_{,\iota}$ can easily derive the structure of the following parts of the communication overlay.

- *Dissemination paths and participants in* O_t: The adversary $\mathfrak{A}\_{,\iota}$ learns the message path that is subsequent to the adversary $\mathfrak{A}\_{,\iota}$. However, based on the routing information, the adversary $\mathfrak{A}\_{,\iota}$ is not able to distinguish brokers and recipients (except in the following case of subjects in leaf-position).
- *Leaf-subjects*: Subjects in leaf-positions are exposed as recipients. Following from the model (see Section 2.2.1), subjects only join a communication overlay, if they are senders or recipients, or are required to relay messages. Thus, subjects in leaf-positions are recipients by design. Additional protection measures can relax this requirement and add usual subjects in leaf positions to the overlay.

With the recipient protection mechanisms (Probabilistic Forwarding and Shell Game [37]; details for both mechanisms are discussed in Section 6.3 in the context of sender and recipient protection) in place, the probability of correctly identifying leaf subjects can be reduced below 0.5. The adversary $\mathfrak{A}\_{,\iota}$ cannot determine the identification probability for subjects located in inner positions of the communication overlay O_t without utilizing additional knowledge. Equation (28) formalizes the probability of an adversary $\mathfrak{A}\_{,\iota}$ identifying a subject v as a recipient based on the information in a successor list. For subjects at the end of the included paths, the probability is less or equal to 0.5 if the recipient protection measures were applied. For the other subjects, an adversary $\mathfrak{A}\_{,\iota}$ cannot provide a conclusive answer.

$$p_{v \overset{?}{\in} \mathcal{R}_t} = \begin{cases} \leqslant 0.5, & d^{-}(v) = 0 \\ ?, & \text{otherwise} \end{cases} \tag{28}$$

The adversary $\mathfrak{A}$ cannot learn information about subjects beyond the direct predecessor. This is caused by the efficiency improving removal of unnecessary ids from the successor list. Thus, the adversary $\mathfrak{A}_{\_,l}$ cannot learn whether the immediate predecessor is the sender or a broker without using additional knowledge, such as another controlled subject on the preceding part of the communication overlay.

5.3.3.2 *Successor Lists with Multi-Layer Encryption*

Using multi-layered encryption, the routing information is encrypted similarly as in OR [62, 124] and mix networks [23]. Each subject on the path can decrypt a single layer of encryption, i.e., can access the very next hop. For that, every subject agrees on a key with each of its neighbors (symmetric encryption) or establishes a public-private key pair and shares the public key with its neighbors (public-key encryption)—the respective keys are then used to encrypt the routing information such that the next subject can access the very next hop.

The encrypted routing information are collected during the overlay establishment, for example in Phase 3 of the ACO mechanism (see Section 4.3.5). For that, subjects perform the following steps:

1. create routing information block ri_{v_i}
   a) recipients: include own id.
   b) brokers b_i: include own id and received *encrypted* (with k_{b_i}) routing information block. Merge multiple received routing information blocks if necessary.
2. forward to next subject towards sender.

Recipients that receive routing information, i.e., recipients that are also brokers, follow also step 1.b) and merge their routing informations as well. The merged encrypted routing information do not reveal whether there is a junction, i.e., there are multiple (partially) disjoint paths included, or just a single but longer path; thus, brokers cannot learn the topology beyond their direct neighbors. The merging, however, requires subjects to cache their routing information blocks. The routing itself is also possible without merging but leaks information about the disjoint paths.

Figure 47 and Example 24 exemplify the how the routing information for the example overlay in Figure 46 will look like. The respective encryption keys are noted with k_x where x denotes the respective subject-id.

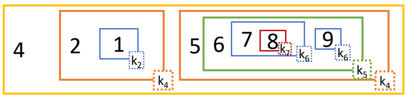


Figure 47: The routing information from the example in Figure 46 using successor lists with multi-layer encryption.

Example 24: The routing information in Figure 47 are established stepwise as follows:

1. round
   8: $\mathtt{ri}_8 = \mathtt{enc}_{k_7}(8)$; forward to subject 7.
2. round
   7: $\mathtt{ri}_7 = \mathtt{enc}_{k_6}(7, \mathtt{enc}_{k_7}(8))$; forward to subject 6.
   9: $\mathtt{ri}_9 = \mathtt{enc}_{k_6}(9)$; forward to subject 6.
3. round
   6: merge $\mathtt{ri}_7$ and $\mathtt{ri}_9$;
   $\mathtt{ri}_6 = \mathtt{enc}_{k_5}(6, \mathtt{enc}_{k_6}(7, \mathtt{enc}_{k_7}(8); 9))$;
   forward to subject 7.
   1: $\mathtt{ri}_1 = \mathtt{enc}_{k_2}(1)$; forward to subject 2.
4. round
   5: $\mathtt{ri}_5 = \mathtt{enc}_{k_4}(5, \mathtt{enc}_{k_5}(6, \mathtt{enc}_{k_6}(7, \mathtt{enc}_{k_7}(8); 9)))$;
   forward to subject 4.
   2: $\mathtt{ri}_2 = \mathtt{enc}_{k_4}(2, \mathtt{enc}_{k_2}(1))$; forward to subject 4.
5. round
   4: store $\mathtt{ri}_5$ and $\mathtt{ri}_2$ to include both in messages towards recipients in O_t.

The sender will use the received routing information block and include it in all messages that are emitted for the respective overlay. When relaying messages, subjects will decrypt their respective layer and can access the next hop. If a subject has to duplicate the message and forward it to two (or more) subjects, it will find multiple encrypted routing blocks and copies the message accordingly. As in the successor list-based mechanism, subjects will only include necessary routing information, i.e., will remove their own id and routing information of other branches.

Proposition 10 *Multi-layered encryption provides a trade-off between efficiency and anonymity by enabling routing (in the limits of the overlay establishment mechanism) with protected routing information where each subject can only see the direct predecessor and direct successors.*

The encryption itself is based upon symmetric encryption using AES with 128 bit key length, which is, according to ENISA [50], assumed to be sufficient for at least ten years. Therefore, the blocksize $\mathrm{enc}_{\mathrm{blocksize}}$ is also 128 bit.

EFFICIENCY Successor lists with multi-layer encryption require memory at every subject v for the key storage, here, for each key is a memory of 128 bit necessary. The memory requirement of each subject is therefore determined by the key size and number of neighbors, on average the required memory is:

$$v.\mathrm{mem} = |sk_{(v,v_k)}| \cdot d(v) \tag{29}$$

To encrypt their id, subjects have to fill a complete encryption block by the requirements of symmetric encryption if necessary they have to add padding to the id. Each encrypted id adds therefore $\lceil |v.\mathrm{address}| / \mathrm{enc}_{\mathrm{blocksize}} \rceil \cdot \mathrm{enc}_{\mathrm{blocksize}}$ bits to the routing information.

Example 25: Using with IPv6 addresses, no additional overhead due to padding arises as the size of the IPv6 addresses (128 bit) matches the blocksize of AES ($\mathrm{enc}_{\mathrm{blocksize}} = 128$ bit). In contrast, when using IPv4 addresses, the id requires only 32 bit. Therefore, each id will be padded to match the 128 bit block size; each id results in an 128 bit cipher text.

	id (size)	SL Entry Size
SL w. MLE	IPv6 (128 bit)	$(\lceil 128\,\mathrm{bit} / 128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 128\,\mathrm{bit}$
	IPv4 (32 bit)	$(\lceil 32\,\mathrm{bit} / 128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 128\,\mathrm{bit}$

The sender has also to save the whole list with routing information; thus, the sender has to provide additionally:

$$\begin{aligned} &s.\mathrm{mem} = v_l.\mathrm{mem} + \\ &\mathcal{V}_t \cdot (\lceil |v_l.\mathrm{address}| / \mathrm{enc}_{\mathrm{blocksize}} \rceil \cdot \mathrm{enc}_{\mathrm{blocksize}}) \end{aligned} \tag{30}$$

Similar to pure successor lists, the communication overhead induced by successor lists with multi-layer encryption on subjects depends on the length of the following path. This communication overhead, in comparison to pure successor lists, is increased by the necessary padding:

$$\begin{aligned} \mathsf{com}_{\mathsf{oh}} = \mathsf{apl}(O_t)\cdot \\ (\lceil |\nu_l.\mathsf{address}| / \mathsf{enc}_{\mathsf{blocksize}} \rceil \cdot \mathsf{enc}_{\mathsf{blocksize}}) \end{aligned} \tag{31}$$

ANONYMITY When the successor list is encrypted with multi-layered encryption, the adversary $\mathfrak{A}_{\_,l}$ is no longer able to decrypt the whole list of succeeding subject ids.

Using the encrypted successor list, the adversary $\mathfrak{A}_{\_,l}$, nevertheless, can learn the following information:

- *Size of the succeeding paths in* O_t: The adversary $\mathfrak{A}_{\_,l}$ can use the knowledge about the identifier size to estimate the number of subjects $\mathsf{numof}(\mathsf{id})$ being contained in the encrypted successor list SL_{enc}, as formalized by Equation (32)

  $$\mathsf{numof}(\mathsf{id}) = \mathsf{sizeof}(SL_{enc}) / \mathsf{sizeof}(\mathsf{id}_{enc}) \tag{32}$$

  where SL_{enc} denotes the encrypted successor list.

Using the size estimation of Equation (32), the adversary $\mathfrak{A}_{\_,l}$ is obviously able to identify subjects in leaf positions if the $\mathfrak{A}_{\_,l}$ is the direct predecessor of the leaf subject. For the other subjects, an adversary $\mathfrak{A}_{\_,l}$ is not able to provide a conclusive answer as they cannot even decide whether a specific subject is part of the communication overlay. Equation (33) formalizes the probability of an adversary of identifying a subject ν as recipient given that the recipient protection measures were applied.

$$p_{\nu \overset{?}{\in} \mathcal{R}_t} = \begin{cases} \leqslant 0.5, & \mathsf{sizeof}(SL_\nu) \approx 0 \\ ?, & \mathsf{otherwise} \end{cases} \tag{33}$$

Here, SL_ν denotes the encrypted (and thus not accessible) remaining part of the successor list that is forwarded towards subject ν—depending on the implementation, this part may be actually non-existent, which leads to the same conclusion.

In contrast to the successor lists described before, this mechanism using multi-layer encryption prevents the adversary from learning about junctions in the communication overlay O_t where messages are duplicated and send towards different paths. Considering the example given in Figure 46, an adversary is not able to distinguish a straight path, for instance, provided by (2–1), from a junction, for example, given by $\{7, 9\}$, if the number of subjects in the considered part of the communication overlay O_t is equal.

Similar to the successor lists without multi-layered encryption, subjects remove their ids to improve efficiency. As a result of this, an adversary $\mathfrak{A}$ cannot learn information about preceding parts of the communication overlay O_t, and, thus, the sender of a message.

5.3.3.3 $\mathcal{V}_t$ *contained in Bloom Filters*

Bloom filters [16, 57] (see Section 2.2.7) can be used to enable the dissemination of routing information. In contrast to the other mechanisms (successor lists (with multi-layer encryption) and distributed routing tables), Bloom filter-based routing is based on a probabilistic data structure. Thus, the occurrence of *false-positives* in the evaluation of the Bloom filter may lead to subjects receiving messages—or relaying messages—even though they are not involved in the communication overlay O_t.

During the overlay establishment, subjects interested in receiving information (the recipients) are including themselves in a Bloom filter and forward the Bloom filter towards the senders. Brokers relay the Bloom filter after including their ids—and, if they are junctions in the communication overlay, merging the Bloom filters as described in Section 2.2.7 using the *or* operation to combine two Bloom filters[1] $\mathbf{bf}_l$ and $\mathbf{bf}_r$ to the resulting Bloom filter $\mathbf{bf}_m$:

$$\mathbf{bf}_m[i] = \begin{cases} 1, & \text{if } \mathbf{bf}_l[i] = 1 \text{ or } \mathbf{bf}_r[i] = 1 \\ 0, & \text{otherwise} \end{cases}$$

Senders will include the established Bloom filter in all messages. Subjects will check if the ids of their neighbors are included in the Bloom filter, and forward the message to all neighbors that are contained in the Bloom filter.

If appropriately configured (see Section 2.2.7), false positive hits are in the worst case "as configured" and falsely forwarded

[1] More than two Bloom filters are merged by applying the or-operator on all Bloom filters that are to be merged.

messages stop after only a few unnecessary hops eventually. Example 26 shows probabilities of messages that are relayed due to consecutive false positive evaluations; it is assumed that the fill level of the Bloom filter is at maximum as high as chosen in the configuration, i.e., the probability of false positives is as expected.

Example 26: Let the Bloom filter be configured to have a false positive probability of $p_{fp} = 0.1$ and 0.01 the average outgoing degree is $\overline{d^-} = 4$. The probability of having n consecutive false positive hits is $(p_{fp} \cdot \overline{d^-})^n$

Consec. False Pos.	$p_{fp} = 0.1$	$p_{fp} = 0.01$
$n = 1$	$0.4^1 = 0.4$	$0.04^1 = 0.04$
$n = 2$	$0.4^2 = 0.16$	$0.04^2 = 0.0016$
$n = 3$	$0.4^3 = 0.064$	$0.04^3 = 0.000064$
$n = 4$	$0.4^4 = 0.0256$	$0.04^4 = 2.56e^{-6}$
$n = 5$	$0.4^5 = 0.01024$	$0.04^5 = 1.024e^{-7}$

Proposition 11 *Bloom filters enable successful routing by providing the subject-ids that are included in the communication overlay. As such, Bloom filters require constant space and provide efficient communication if configured appropriately. Bloom filters provide anonymity due to the non-invertible property of the used hash functions.*

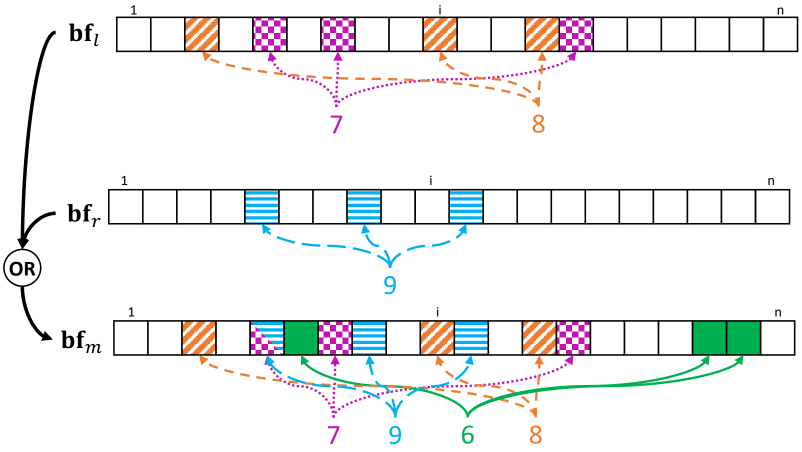


Figure 48: Bloom filter-based routing information with merging of Bloom filters at subject 6; Subjects 7 and 8 are hashed into bf_l; subject 9 is hashed into bf_r. Subject 6 merges both bf_l and bf_r to establish bf_m and hashes their own id.

EFFICIENCY Using Bloom filters, subjects add their id to a Bloom filter when they receive such a Bloom filter during overlay establishment. To establish these Bloom filters, recipients create a Bloom filter and forward it to their predecessor. Intermediate junction-subjects merge the Bloom filters received in the same round; otherwise, they will simply forward the Bloom filter; brokers and recipients do not store the Bloom filters. Senders merge all received Bloom filters and create the Bloom filter that comprises all necessary routing information to relay messages to all connected recipients.

The Bloom filter configuration determines the expectable rate of false positives; the probability of false positive depends on the number of hash functions, the size of the Bloom filter, and the number of entries added to the Bloom filter. The configured size of the Bloom filter depends on the number of expected subjects that are required to establish the communication overlay—and, thus, their ids are added to the Bloom filter.

For the estimation of m_{bf}, an *aggressive* strategy can be used that directly estimates $|\mathcal{V}_t|$ to derive the necessary Bloom filter size m_{bf}, for example, by using precomputed tables as given in Table C.30. A more *conservative* strategy adds a buffer to the estimate before deriving m_{bf} and not use the estimated size of $\mathcal{V}_t$ directly. The size of this buffer depends on the impact of increased communication overhead on the system; if the system is sensitive to communication overhead, the size m_{bf} of the Bloom filter may even be estimated using $\mathcal{V}$ instead of $\mathcal{V}_t$.

With that, the configuration also determines the memory demand and communication overhead.

The memory demand differs between senders and the other subjects. Only senders $s \in \mathcal{S}_t$ have to store the Bloom filter:

$$v.mem = \begin{cases} m_{bf}, & v = s \in \mathcal{S}_t \\ 0, & \text{otherwise} \end{cases} \tag{34}$$

The communication overhead is stable with the size m_{bf} of the Bloom filter as the Bloom filter is not flexible and depending on the actual number of added ids. Even though some Bloom filter variants (for example, Fan et al. [57]) enable deletion of entries, the size m_{bf} of the Bloom filter itself does not change.

Example 27: Using an exemplary Bloom filter configuration with three hash functions, communication overlay sizes $|\mathcal{V}_t|$ of 8 (see example in Figure 46) and 60 (see communication overlay sizes in Section 4.4.5), and acceptable false positive probabilities p_{fp} of 0.1 and 0.01, the Bloom filter has to have the following sizes. In this example, the conservative strategy adds a buffer of 30 %.

	n_{bf} estimation	$p_{fp} = 0.1$	$p_{fp} = 0.01$
aggressive	$\mathcal{V}_t = 8$	40 bit	104 bit
	$\mathcal{V}_t = 60$	300 bit	780 bit
conservative	$\mathcal{V}_t = 8 \cdot 1.3 = 11$	55 bit	143 bit
	$\mathcal{V}_t = 60 \cdot 1.3 = 78$	390 bit	1,014 bit

A precomputed table of Bloom filter sizes, added entries, the number of hash functions and the resulting false positive probabilities is given in Appendix C.

ANONYMITY Including the subject ids in a Bloom filter prevents the adversary $\mathfrak{A}_{\_,\iota}$ from learning structural information about O_t. If the adversary $\mathfrak{A}_{\_,\iota}$ is aware of the id-space, they are able to check the ids using brute force; this, however, also yields the blur of false positive hits.

When evaluating the routing information contained in the Bloom filter, the adversary $\mathfrak{A}_{\_,\iota}$ can learn the following information:

- *Subjects* $\mathcal{V}_t$ *of* O_t: By evaluating the routing information using brute force, the adversary $\mathfrak{A}_{\_,\iota}$ can derive $\mathcal{V}_t$. However, the connections $\mathcal{E}_t$ are not derivable using this technique and are, therefore, out of reach of such an adversary $\mathfrak{A}_{\_,\iota}$ without access to additional information, for example, a global view on the system.

- *Sender* s: The adversary $\mathfrak{A}_{\_,\iota}$ can identify a sender s relying on strong assumptions:

  1. The sender s does not include their id to the Bloom filter and
  2. The adversary $\mathfrak{A}$ is a direct successor of the sender s.

  If both requirements are fulfilled, the adversary $\mathfrak{A}_{\_,\iota}$ can learn that the id of their predecessor is not included in the Bloom filter, even though the subject handled the message. Accounting for the false positive probability, the adversary

$\mathfrak{A}_{\_\iota}$ learns that their predecessor ν is sender s with $p_{\nu \overset{?}{\in} \mathcal{S}_t} = 1 - p_{fp}$.

Beyond this information, an adversary $\mathfrak{A}_{\_\iota}$ is not able to learn additional information about the communication overlay O_t.

5.3.3.4 *Subjects maintaining Distributed Lookup Tables*

Distributed lookup tables enable subjects to act as brokers without having routing information, besides a topic-id and message-id, included in the messages. Using lookup tables, subjects maintain their own limited view of the system and the overlay.

Subjects add successors and predecessor to their local lookup table when the subject learns about a topic and the respective interests of sending or receiving messages for this topic. These tables manifest the distributed knowledge on the topological structure of the communication overlay.

Senders s include the topic identifier t in each message that they disseminate in the communication overlay O_t. Upon receiving a message, subjects try to map the included topic identifier to their neighbors using their local lookup table. The message is then relayed to all entries that match the topic identifier.

Example 28: Subjects 4 and 6 maintain the following routing tables, for the specified topic t (see Figure 46) and the additional, not further specified topic t_2.

Topic	Direction	$\nu = 4$
t	Pre	$\{\}$
	Suc	$\{2, 5\}$
t_2	Pre	$\{5\}$
	Suc	$\{\}$

Topic	Direction	$\nu = 6$
t	Pre	$\{5\}$
	Suc	$\{7, 9\}$
t_2	Pre	$\{5\}$
	Suc	$\{7\}$

Proposition 12 *Distributed lookup tables establish routing information by memorizing successors and predecessors at the subjects itself. Routing information do not need to be included in messages, yet, shared knowledge ensures successful routing, while improving anonymity as information is not repeatedly spread in the system.*

EFFICIENCY Using distributed lookup tables, subjects keep the routing information locally; messages are associated with communication overlays by including the topic identifier t.

During the overlay establishment, subjects establish their routing tables by storing topic identifiers t with their respective predecessors and successors to be able to relay messages. With that, the memory demand of each subject is bound by the number of direct neighbors and the number of topics in T:

$$v.mem_{max} = d^{-}(v) \cdot |v.address| \cdot |T| \tag{35}$$

As messages do not include routing information, the communication overhead is determined by the size of the topic identifier itself:

$$com_{oh} = |t_i| \tag{36}$$

There is no differentiation of requirements for different roles necessary; all subjects have to maintain the routing information based on their neighborhood.

ANONYMITY With distributed lookup tables, all subjects keep their routing information locally. The adversary $\mathfrak{A}_{\_,l}$ is not able to learn information from these tables as they are not intended to be shared.

Without additional information about the communication overlay O_t, the adversary $\mathfrak{A}_{\_,l}$ cannot even identify leaf-subjects when they are directly adjacent to the adversary.

5.3.4 *Qualitative Discussion of Routing Information Exchanges*

This section analyzes the different routing information exchange mechanisms in detail using a detailed qualitative discussion. The discussion focusses on the adversary presented in 5.3.2. First, efficiency is discussed from the perspective of communication overhead in message size as well as memory overhead at the subjects. Second, the influence on anonymity is discussed by analyzing the information leakage towards the adversary.

5.3.4.1 *Efficiency: Communication & Memory Overhead*

Efficiency depends on the characteristics of the specific mechanisms to exchange routing information but may also depend on the properties of the system. For that, this section discusses the communication and memory overhead for three exemplary system manifestations:

1. (EX) *Example given in Figure 46*
   - $|\mathcal{V}|=9$, $|\mathcal{R}_t|=4$, $|\mathcal{V}_t|=8$
   - $apl(O_t)=3$, $\overline{d(v)}=2.4$
2. (SOC-F) *Social network with* few *recipients, based on Section 4.4.4.2 with* $rr = 0.015$
   - $|\mathcal{V}|=2,000$, $|\mathcal{R}_t|\approx 30$, $|\mathcal{V}_t|\approx 62$
   - $apl(O_t)=5$, $\overline{d(v)}=4$
3. (SOC-M) *Social network with* many *recipients, based on Section 4.4.4.2 with* $rr = 0.05$
   - $|\mathcal{V}|=2,000$, $|\mathcal{R}_t|\approx 100$, $|\mathcal{V}_t|\approx 161$
   - $apl(O_t)=6$, $\overline{d(v)}=4$
4. (SOC2-F) *Larger social network with* few *recipients, based on Section 4.4.1 with* $\mathcal{V} = 4,000$ *and* $rr = 0.0075$
   - $|\mathcal{V}|=4,000$, $|\mathcal{R}_t|\approx 30$, $|\mathcal{V}_t|\approx 70$
   - $apl(O_t)=6$, $\overline{d(v)}=4$
5. (SOC2-M) *Larger social network with* many *recipients, based on Section 4.4.4.2 with* $\mathcal{V} = 4,000$ *and* $rr = 0.025$
   - $|\mathcal{V}|=4,000$, $|\mathcal{R}_t|\approx 100$, $|\mathcal{V}_t|\approx 181$
   - $apl(O_t)=6$, $\overline{d(v)}=4$

The remaining relevant properties are assumed to be:

- IPv6 as ids: $|v.address| = 128\,\text{bit}$
- Bloom filter: $p_{fp} = 0.01$; $k_{bf} = 3$
- $|t| = 12\,\text{bit}$; $|T|_{max} = 4,096$

COMMUNICATION OVERHEAD

- Successor lists do not add additional information beyond the plain ids. Using Equation (27) ($com_{oh} = apl(O_t) * |v.address|$), the following communication overheads result for the five example networks:
  1. $com_{oh}^{EX} = 1.5 \cdot 128\,\text{bit} = 192\,\text{bit}$
  2. $com_{oh}^{SOC-F} = 2.5 \cdot 128\,\text{bit} = 320\,\text{bit}$
  3. $com_{oh}^{SOC-M} = 3 \cdot 128\,\text{bit} = 384\,\text{bit}$

4. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-F}} = 3 \cdot 128\,\mathrm{bit} = 384\,\mathrm{bit}$
5. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-M}} = 3 \cdot 128\,\mathrm{bit} = 384\,\mathrm{bit}$

- Successor lists with multi-layer encryption add padding through the encryption of the ids and may, therefore, add further overhead. Using Equation (31) ($\mathrm{com}_{\mathrm{oh}} = \mathrm{apl}(O_t) \cdot (\lceil |v_l.\mathrm{address}|/\mathrm{enc}_{\mathrm{blocksize}} \rceil \cdot \mathrm{enc}_{\mathrm{blocksize}})$), the following communication overheads result for the five example networks:

  1. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{EX}} = 1.5 \cdot (\lceil 128\,\mathrm{bit}/128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 192\,\mathrm{bit}$
  2. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-F}} = 2.5 \cdot (\lceil 128\,\mathrm{bit}/128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 320\,\mathrm{bit}$
  3. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-M}} = 3 \cdot (\lceil 128\,\mathrm{bit}/128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 384\,\mathrm{bit}$
  4. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-F}} = 3 \cdot (\lceil 128\,\mathrm{bit}/128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 384\,\mathrm{bit}$
  5. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-M}} = 3 \cdot (\lceil 128\,\mathrm{bit}/128\,\mathrm{bit} \rceil \cdot 128\,\mathrm{bit}) = 384\,\mathrm{bit}$

- Bloom filters are configured with an expectation of the required size to meet a specific false positive probability (see Section 2.2.7); the size of a Bloom filter remains stable independent from the number of added entries. For the *aggressive* strategy, which configures the size of the Bloom filter with the size of the overlay directly, the following communication overhead result for the five example networks:

  1. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{EX}} = 104\,\mathrm{bit}$
  2. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-F}} = 806\,\mathrm{bit}$
  3. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-M}} = 2,093\,\mathrm{bit}$
  4. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-F}} = 910\,\mathrm{bit}$
  5. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-M}} = 2,353\,\mathrm{bit}$

  For the *conservative* strategy, which configures the size of the Bloom filter with the size of the overlay and adds 30% buffer, the following communication overhead result for the five example networks:

  1. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{EX}} = 143\,\mathrm{bit}$
  2. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-F}} = 1,053\,\mathrm{bit}$
  3. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC-M}} = 2,730\,\mathrm{bit}$
  4. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-F}} = 1,183\,\mathrm{bit}$
  5. $\mathrm{com}_{\mathrm{oh}}^{\mathrm{SOC2-M}} = 3,059\,\mathrm{bit}$

- Using distributed lookup tables, subjects keep routing information locally. To enable the routing, messages carry a topic identifier t_i.
  1. $com_{oh}^{EX} = 12\,\text{bit}$
  2. $com_{oh}^{SOC-F} = 12\,\text{bit}$
  3. $com_{oh}^{SOC-M} = 12\,\text{bit}$
  4. $com_{oh}^{SOC2-F} = 12\,\text{bit}$
  5. $com_{oh}^{SOC2-M} = 12\,\text{bit}$

MEMORY OVERHEAD

- Successor lists are stored at the senders s; other subjects v are not required to hold memory. Therefore, the memory requirements for usual subjects $v.mem$ for the three example networks are:
  1. $v.mem^{EX} = 0\,\text{bit}$
  2. $v.mem^{SOC-F} = 0\,\text{bit}$
  3. $v.mem^{SOC-M} = 0\,\text{bit}$
  4. $v.mem^{SOC2-F} = 0\,\text{bit}$
  5. $v.mem^{SOC2-M} = 0\,\text{bit}$

  As senders have to store the successor list, they have to hold memory for the list. Their memory requirement can be derived using Equation (26) ($s.mem = \mathcal{V}_t \cdot |v.address| - |v.address|$):
  1. $s.mem^{EX} = 8 \cdot 128\,\text{bit} - 128\,\text{bit} = 896\,\text{bit}$
  2. $s.mem^{SOC-F} = 62 \cdot 128\,\text{bit} - 128\,\text{bit} = 7,808\,\text{bit}$
  3. $s.mem^{SOC-M} = 161 \cdot 128\,\text{bit} - 128\,\text{bit} = 20,480\,\text{bit}$
  4. $s.mem^{SOC2-F} = 70 \cdot 128\,\text{bit} - 128\,\text{bit} = 8,832\,\text{bit}$
  5. $s.mem^{SOC2-M} = 181 \cdot 128\,\text{bit} - 128\,\text{bit} = 23,040\,\text{bit}$

- Successor lists with multi-layer encryption require subjects v to establish keys with all neighbors. Using Equation (29) ($v.mem = |sk_{(v,v_k)}| \cdot d(v)$), the memory requirements for usual subjects $v.mem$ have to be met for the three example networks:
  1. $v.mem^{EX} = 2.4 \cdot 128\,\text{bit}) = 307.2\,\text{bit}$
  2. $v.mem^{SOC-F} = 4 \cdot 128\,\text{bit}) = 512\,\text{bit}$

3. $v.mem^{SOC-M} = 4 \cdot 128\,\text{bit}) = 512\,\text{bit}$
4. $v.mem^{SOC2-F} = 4 \cdot 128\,\text{bit}) = 512\,\text{bit}$
5. $v.mem^{SOC2-M} = 4 \cdot 128\,\text{bit}) = 512\,\text{bit}$

Senders also have to store the encrypted successor list, their memory requirement can be derived using Equation (30) ($s.mem = v.mem + \mathcal{V}_t \cdot (\lceil |v.address|/enc_{blocksize} \rceil \cdot enc_{blocksize})$):

1. $s.mem^{EX} = 307.2\,\text{bit} + 8 \cdot (\lceil 128\,\text{bit}/128\,\text{bit} \rceil \cdot 128\,\text{bit}) = 1,344\,\text{bit}$
2. $s.mem^{SOC-F} = 512\,\text{bit} + 62 \cdot (\lceil 128\,\text{bit}/128\,\text{bit} \rceil \cdot 128\,\text{bit}) = 8,448\,\text{bit}$
3. $s.mem^{SOC-M} = 512\,\text{bit} + 161 \cdot (\lceil 128\,\text{bit}/128\,\text{bit} \rceil \cdot 128\,\text{bit}) = 21,120\,\text{bit}$
4. $s.mem^{SOC2-F} = 512\,\text{bit} + 70 \cdot (\lceil 128\,\text{bit}/128\,\text{bit} \rceil \cdot 128\,\text{bit}) = 9,472\,\text{bit}$
5. $s.mem^{SOC2-M} = 512\,\text{bit} + 181 \cdot (\lceil 128\,\text{bit}/128\,\text{bit} \rceil \cdot 128\,\text{bit}) = 23,680\,\text{bit}$

- Bloom filters are stored only at the sender subjects s; other subjects v are therefore not required to provide memory:

  1. $v.mem^{EX} = 0\,\text{bit}$
  2. $v.mem^{SOC-F} = 0\,\text{bit}$
  3. $v.mem^{SOC-M} = 0\,\text{bit}$
  4. $v.mem^{SOC2-F} = 0\,\text{bit}$
  5. $v.mem^{SOC2-M} = 0\,\text{bit}$

  For the *aggressive* strategy, senders s have to hold the following memory for the three example networks:

  1. $s.mem^{EX} = 104\,\text{bit}$
  2. $s.mem^{SOC-F} = 806\,\text{bit}$
  3. $s.mem^{SOC-M} = 2,093\,\text{bit}$
  4. $s.mem^{SOC2-F} = 910\,\text{bit}$
  5. $s.mem^{SOC2-M} = 2,353\,\text{bit}$

  For the *conservative* strategy, senders s have to hold the following memory for the three example networks:

  1. $s.mem^{EX} = 143\,\text{bit}$
  2. $s.mem^{SOC-F} = 1,053\,\text{bit}$

3. $s.mem^{SOC-M} = 2{,}730\,\text{bit}$
4. $s.mem^{SOC2-F} = 1{,}183\,\text{bit}$
5. $s.mem^{SOC2-M} = 3{,}059\,\text{bit}$

- Using distributed lookup tables, subjects have to maintain their routing information locally. As such, all subjects have to hold the following memory *per topic* $t \in T$ which is also described by Equation (35):
    1. $v.mem^{EX} = 2.4 \cdot 128\,\text{bit} = 307.2\,\text{bit}$
    2. $v.mem^{SOC-F} = 4 \cdot 128\,\text{bit} = 512\,\text{bit}$
    3. $v.mem^{SOC-M} = 4 \cdot 128\,\text{bit} = 512\,\text{bit}$
    4. $v.mem^{SOC2-F} = 4 \cdot 128\,\text{bit} = 512\,\text{bit}$
    5. $v.mem^{SOC2-M} = 4 \cdot 128\,\text{bit} = 512\,\text{bit}$

Table 22 summarizes the results of the efficiency properties of the four mechanisms to exchange routing information on the five exemplary networks. An important finding is that the memory requirements seem to depend on the size of the communication overlay but not on the size of the communication system itself. Only the successor lists with multi-layer encryption and the distributed lookup tables require all subjects to hold memory—the memory requirements depend on the size of the neighborhood of the subjects in the communication system. The size of the Bloom filters depend on the size of the communication overlay; larger overlays require larger Bloom filters and therefore more memory. Smaller Bloom filters still enable routing. However, the false positive ratio will increase and cause messages that are forwarded to subjects that are not part of the overlay.

5.3.4.2 *Anonymity*

The adversary $\mathfrak{A}_{\_,l}$ may collect evidence from the routing information that increases their probability of identifying sender and recipients of a message. Also, the adversary $\mathfrak{A}_{\_,l}$ may be able to derive the topology of the communication overlay from the routing information.

SENDER AND RECIPIENT IDENTIFICATION Section 4.4.6 pointed out that an adversary $\mathfrak{A}_{p,g}$ can identify subjects in leaf

Table 22: Comparison of efficiency properties of routing information exchange mechanisms.

Measure	Network	SL	SL_{enc}	$BF_{aggr.}$	$BF_{cons.}$	DLT
$\mathtt{com}_{oh}$	EX	192 bit	192 bit	104 bit	143 bit	12 bit
	SOC-F	320 bit	320 bit	806 bit	1,053 bit	12 bit
	SOC-M	384 bit	384 bit	2,093 bit	2,730 bit	12 bit
	SOC2-F	384 bit	384 bit	910 bit	1,183 bit	12 bit
	SOC2-M	384 bit	384 bit	2,353 bit	3,059 bit	12 bit
$\mathtt{v.mem}$	EX	0 bit	307.2 bit	0 bit	0 bit	307.2 bit
	SOC-F	0 bit	512 bit	0 bit	0 bit	512 bit
	SOC-M	0 bit	512 bit	0 bit	0 bit	512 bit
	SOC2-F	0 bit	512 bit	0 bit	0 bit	512 bit
	SOC2-M	0 bit	512 bit	0 bit	0 bit	512 bit
$\mathtt{s.mem}$	EX	896 bit	1,344 bit	104 bit	143 bit	0 bit
	SOC-F	7,808 bit	8,448 bit	806 bit	1,053 bit	0 bit
	SOC-M	20,480 bit	21,120 bit	2,093 bit	2,730 bit	0 bit
	SOC2-F	8,832 bit	9,472 bit	910 bit	1,183 bit	0 bit
	SOC2-M	23,040 bit	23,680 bit	2,353 bit	3,059 bit	0 bit

positions as senders or recipients. Even with the recipient protection measures, for example, shell game and probabilistic forwarding [37], leaf positions leave evidence. The different mechanisms to exchange routing information enable routing differently; therefore, the leaked information to the adversary $\mathfrak{A}_{\_,\iota}$ differs when the routing information is accessed:

- *Successor Lists* include the routing information in messages as a list of subject ids. When the adversary $\mathfrak{A}_{\_,\iota}$ accesses such a successor list, they can directly derive the leaf positions by learning the subjects at which the paths in the successor list end. As a result of this, the adversary $\mathfrak{A}_{\_,\iota}$ can collect evidence on recipients in leaf positions.

  As subjects remove their own id from the successor list before forwarding the message, the adversary $\mathfrak{A}_{\_,\iota}$ cannot collect evidence beyond their direct predecessor. Therefore, the adversary $\mathfrak{A}_{\_,\iota}$ cannot provide a conclusive answer about the sender of a message.

- *Successor Lists with Multi-Layer Encryption* include the routing information as a list of subject ids with multiple layers of encryption. As such, the adversary $\mathfrak{A}_{\_,\iota}$ cannot derive subject ids beyond their direct neighbors. As a result of this, the adversary can also no longer collect evidence on leaf subjects, and therefore the adversary cannot collect evidence on recipients in leaf positions. However, the adversary $\mathfrak{A}_{\_,\iota}$ is still able to estimate the number of subjects that follow their direct successor— with this, the adversary $\mathfrak{A}_{\_,\iota}$ can, of course, identify immediate successors that do not further forward the message.

Similar to simple successor lists, subjects strip their own ids from the successor list. As a result of this, the adversary $\mathfrak{A}_{\_,\iota}$ cannot collect evidence beyond their direct predecessor. Therefore, the adversary $\mathfrak{A}_{\_,\iota}$ cannot provide a conclusive answer about the sender of a message.

- *Bloom filter*-based routing uses a Bloom filter which contains the subject ids of subjects that participate in the communication overlay. While the adversary $\mathfrak{A}_{\_,\iota}$ may use a brute force approach to identify all subjects that participate in the communication overlay (including possible false positives), this information does not provide evidence about leaf positioned recipients.

  Similar to the case for recipients, the information about participation in the overlay itself does not yield evidence that may enable the adversary $\mathfrak{A}_{\_,\iota}$ to provide a conclusive answer about the sender of a message. In an edge case, the adversary $\mathfrak{A}_{\_,\iota}$ can identify the sender: the sender has to be the direct predecessor of the adversary $\mathfrak{A}_{\_,\iota}$ and must not hash their own id into the Bloom filter. Then, the adversary $\mathfrak{A}$ learns that the Bloom filter does not contain the subject that sent the message; the only case that allows this is that the sending subject is the actual sender of the message.

- *Distributed Lookup Tables* do not distribute any routing information using the messages; the knowledge is kept within the 1-hop neighborhood of a subject. As a result of this, the adversary $\mathfrak{A}$ cannot collect any evidence about subjects that are located behind their direct successors, or predecessors respectively. The adversary $\mathfrak{A}$ cannot provide a conclusive answer about senders or recipients based on the knowledge of their own distributed lookup table.

TOPOLOGY LEARNING Learning the topology, i.e., learning the participants of an overlay and their respective connections in the overlay, may enable the adversary $\mathfrak{A}\_,\iota$ to prepare further attacks even though it might not help to identify sender and recipients of a message directly.

The mechanisms to exchange routing information leak different information about the participants of an overlay and their respective connections. Obviously, the adversary $\mathfrak{A}\_,\iota$ is aware of their direct predecessor as well as of their direct successors. Beyond that, the information leakage is as follows:

- *Successor Lists* enable the adversary $\mathfrak{A}\_,\iota$ to learn the whole subsequent dissemination tree. As such, the adversary $\mathfrak{A}\_,\iota$ learns the ids of all succeeding subjects as well as their connections in the communication overlay.

  Towards the sender of a message, successor lists do not include any information. As a result of this, the evidence that the adversary $\mathfrak{A}\_,\iota$ collects depends strongly on their position in the communication overlay—the closer the position to the sender, the more information is accessible.

- *Successor Lists with Multi-Layer Encryption* enable the adversary $\mathfrak{A}$ to estimate the size of the overlay behind their position. This does, however, not leak information about the participants or their connections.

- *Bloom filters* enable the adversary $\mathfrak{A}\_,\iota$ to check every possible subject id and thereby derive the ids of the participants in the overlay. Hereby, the adversary will also learn the false positives as participants of the communication overlay.

  The adversary $\mathfrak{A}\_,\iota$ cannot collect information about the connections in the overlay as the Bloom filters carry only subject ids but not information about the order of their adding or the connections of the subjects.

- *Distributed Lookup Tables* keep the routing information local with the subjects; messages carry only the topic id. The adversary $\mathfrak{A}\_,\iota$ cannot collect any information about the participants of the overlay or their connections.

Table 23 summarizes the results of the mechanisms for routing information exchange with respect to protection of senders and recipients in leaf positions as well as protection of overlay participants and of the respective connections. Bloom filters and distributed lookup tables have the advantage as they limit the information leakage by removing all connectivity information or keeping the information locally at the subjects. Successor lists with multi-layer encryption leak information about the number of subjects in the overlay and about recipients in edge cases. Simple successor lists leak information about recipients in leaf positions as well as information about the overlay behind the adversary $\mathfrak{A}_{\_,\iota}$.

Table 23: Comparison of efficiency properties of routing information exchange mechanisms.

Property	SL	$\mathrm{SL}_{\mathrm{enc}}$	BF	DLT
Sender Protection	✓	✓	✓	✓
Recipient Protection	✗	✢	✓	✓
Participant Protection	✢	✓	✗	✓
Connection Protection	✢	✓	✓	✓

5.3.4.3 *Summary*

Exchange of routing information is in the tension of the conflicting goals efficiency and anonymity. This section introduced and discussed four methods, namely successor lists, successor lists with multi-layer encryption, Bloom filter-based routing information, and distributed lookup tables.

Successor lists are an easy mechanism to exchange routing information. While they are easy to use, their communication overhead increases with the depth of the communication overlay. Similarly, the sender has to hold more memory to store the successor list when the overlay size increases. By including the path information into the message, an adversary $\mathfrak{A}_{\_,\iota}$ can collect evidence on the subsequent overlay and recipients in leaf positions.

Successor lists with multi-layer encryption limit the information leakage about the subsequent communication overlay. However, to enable the encryption, subjects have to establish keys with all direct neighbors. Apart from that, the properties are similar to the simple successor lists: the communication overhead remains stable and depends on the depth of the dissemination tree; the same holds for the memory requirements of the sender.

Additionally, to the memory requirements of all subjects, there is also the computational overhead of the encryption itself.

Bloom filter-based exchange of routing information requires no memory at subjects in the overlay. The sender has to hold the memory to store the Bloom filter; the size of the Bloom filter also defines the communication overhead. The size of the Bloom filter itself depends on the (estimated) size of the communication overlay and the acceptable false positive ratio. The Bloom filter also hides senders, recipients, the connections between the subjects in the overlay, and the order of their adding. With that, the adversary $\mathfrak{A}_{\_,\iota}$ cannot obtain evidence about the senders and recipients.

Distributed lookup tables maintain the routing information local at every subject. As a result of this, the memory requirements depend on the size of the neighborhood of the subjects and the number of topics—more specifically on the number of topics in which a subject is involved. The communication overhead is constant with the size of the topic id of the message and is independent of the size of the communication system and the size of the communication overlay. The adversary $\mathfrak{A}_{\_,\iota}$ cannot learn information beyond their direct neighbors as subjects to not exchange any routing information.

5.4 CONCLUSION

This chapter reasoned about the challenge of dynamic subjects and routing in the tension between efficiency and anonymity. For that, this chapter presented and discussed mechanisms of the novel ACO-based overlay establishment mechanism that facilitated the establishment of communication in dynamic environments with leaving and joining subjects. After that, an in-depth discussion of different mechanisms to exchange routing information is described.

Dynamic subjects, i.e., subjects that may leave, fail, and join (*churn*) the communication system at any point in time, require mechanisms to handle disruptions in communication overlays. The temporal behavior of subjects is described using long-tailed distributions, e.g., the Weibull distribution [66, 137]. Defining session durations and intersession times of subjects according to a Weibull distribution, a simulation study with dynamic subjects was performed to show that the utilization of pheromones to decrease the reconnection time of subjects in the system to the communication overlay. The simulation study has shown that the ACO-based overlay establishment mechanism with its techniques to counter disruptions can provide higher reliability towards the active subjects in a communication compared to traditional flooding-based shortest path communication overlays.

Subject churn requires frequent exchanges of routing information. This information exchange reveals the tension between efficiency and anonymity. The more information is available to subjects, the better the routing decisions; as a result of these better routing decisions, the efficiency is improved. However, these efficiency improvements impede anonymity. The available information has to be assumed to be available to an adversary $\mathfrak{A}$, and, as a result of this leaks information to identify active subjects. Besides the information leakage, also the communication overhead due to the amount of shared routing information and the memory overhead to store the shared routing information have to be considered. The appropriate routing information exchange mechanism strongly depends on the application scenario. In a computer- or smartphone-based application scenario, memory as well as connection are usually no limiting factors and provide more freedom to vary the routing information exchange mechanism. In a WSN-based application scenario, both are limited, and the selection has to consider, for example, the population.

The key lessons of this chapter are as follows:

- Dynamic subjects impair communication reliability by disrupting the communication overlay.

- ACO-based communication overlay mechanism can utilize previously established pheromones to improve the reconnection speed and probability.

- Leaving and joining subjects require a frequent exchange of routing information

- Routing information exchange challenges the conflicting goals efficiency and anonymity. More available routing information improve efficiency but impede anonymity

- Different routing information exchange mechanisms differ in their communication overhead, memory overhead, and information leakage. The appropriate mechanism depends on the application scenario.

6 EFFICIENT AND EFFECTIVE SENDER PROTECTION

This chapter introduces a novel mechanism to protect senders from being identified by an adversary. The sender protection mechanism is based upon the anonymization primitive DCnet [22]. The proposed *asymmetric DCnets (ADCnets)* provide means for an adjustable trade-off between the conflicting goals efficiency and anonymity.

This chapter is structured as follows: after providing a motivation in Section 6.1, an extended model and terminology is introduced in Section 6.2 which extend the general model from Section 2.2 with features that are only relevant for this chapter. Section 6.3 addresses the necessity for additional sender protection by showing topological weaknesses of group communication systems and discussing existing protection measures for recipients. In Section 6.4, the usage of the anonymization primitive *cover traffic* for sender protection, the approach to improving its efficiency, and the results of this approach are discussed. The mechanism of ADCnets is then introduced in Section 6.5. Lastly, this chapter is concluded in Section 6.6

6.1 MOTIVATION

Communication overlays O_t often leak information about *some* senders $s \in \mathcal{S}_t$ and recipients $r \in \mathcal{R}_t$. By definition, only active subjects, i.e., senders and recipients, join the communication overlay O_t initially. This is usually not sufficient to connect all senders with all recipients; therefore, a subset of the remaining subjects may join as brokers $b \in \mathcal{B}_t$ to establish a connected overlay. Following this design, the positions in the communication overlay O_t hold information about the role of a subject: the root is the sender, leaves are recipients, and "inner" positions are brokers that may also be recipients. A global adversary $\mathfrak{A}_{p,g}$ like the service provider can derive the communication overlay by tracing the message flow. As a result of this, the adversary $\mathfrak{A}_{p,g}$ can identify subjects in endpoint positions, i.e., senders in root positions and recipients in leaf positions. Thus, anonymity for senders and recipients requires additional protection measures.

6.2 MODEL AND TERMINOLOGY

In the following, this section extends the basic model and terminology given in Chapter 2.

6.2.1 *Cover Traffic and Noise*

Cover traffic composes cover messages m^{cov} with noise that is cover messages m^{cov} contain random bits. The random content ensures that an adversary $\mathfrak{A}_{p,g}$ cannot distinguish noise and encrypted messages.

Sender protection is usually based on noise, may it be in form of cover messages as in the anonymization primitive cover traffic (see Sections 2.4.2.5 and 6.4) or in the form of cryptographic messages of DCnet participants (see Sections 2.4.2.6 and 6.5) that are necessary to decrypt the actual communication messages. Both have in common that an adversary $\mathfrak{A}_{p,g}$, which can observe the message flow in the communication system, cannot distinguish communication messages m^{com} from noise m^{cov}.

6.2.2 *One-Time Pad*

One-time pads provide *perfectly secure* encryption: the XOR-result of a message m_t^{com} with a pad $otp = \mathsf{GF}(2)^n$ of the same length n that is truly random appears to be random as well [131, 132]. The pad otp is used as encryption—and decryption—key. With that, the adversary does not learn anything from intercepting a message that is being encrypted with a one-time pad as all possible decryptions are equally probable with the correct decryption of the message. Only the originally used pad can decrypt the message m_t^{com}.

In the following DCnet-based anonymization scheme, a group of subjects uses their pair-wise keys as one-time pads. Only with all *used* keys at hand, a subject can decrypt the message. Following from requirement, all subjects in the DCnet-subgroup have to contribute their share in the form of their respective keys.

6.2.3 *Dining-Cryptographer Networks*

As described in Section 2.4.2.6, DCnets are originally proposed by Chaum [22].

A DCnet connects an arbitrary set of subjects $\mathcal{V}$ and enables them to share information with provable anonymity. The communication in DCnet is protected by using a form of *secure multi-party computation*

6.2.3.1 *Keys Establishment*

To establish anonymous communication, each subject has to establish keys with a set of neighbors:

$$rk_{k,l} = rk_{l,k}\colon \text{ shared key of subjects } v_k \text{ and } v_l$$

The set of neighbors with which keys are established depends on the underlying key graph for which Chaum proposed four options:

- *complete graph*: in a complete key graph (i.e., a clique), a subject v_k establishes keys with every other subject $v_l \in \mathcal{V}$. In a complete graph, an adversary $\mathfrak{A}_{\_,g}$ will always see the set of non-controlled subjects $\mathcal{AS} = \mathcal{V} \setminus \mathfrak{C}$ as anonymity set.

- *ring*: in a ring-based key graph, a subject v_k establishes keys with their two adjacent subjects v_{k-1} and v_{k+1}. This key-sharing construct enables an adversary $\mathfrak{A}_{a,l}$ or $\mathfrak{A}_{a,c}$ to easily partition the ring; with multiple controlled subjects, i.e., a colluding adversary $\mathfrak{A}_{a,c}$, the adversary can easily de-anonymize the participant that is immediately between them by establishing two anonymity sets $\mathcal{AS}_1$ and $\mathcal{AS}_2$ where $\mathcal{AS}_1$ contains only the suspected subject.

- *subclique*: in a subclique key graph, s subset of subjects forms a clique that ensures the anonymization of the communication. The remaining subjects are attached to this clique and share a key with each of the subjects in the clique. The anonymity towards an adversary $\mathfrak{A}$ is formed by the set of subjects of the anonymization clique that is not under their control $\mathcal{AS} = \mathcal{V}_{DC} \setminus \mathfrak{C}$: as long as there is at least one subject $v_k \in \mathcal{V}_{DC}$ that is not under the adversary's control, i.e., if there is at least one subject $v_k \notin \mathfrak{C}$, it is impossible for the adversary to distinguish subjects sending cover messages m^{cov} from actual senders s_i sending communication messages m_t^{com}. As the anonymization is solely based on the clique $\mathcal{V}_{DC}$, the communication does not rely on contributions of other subjects that are not part of $\mathcal{V}_{DC}$. Dissent [30] establishes anonymous communication based on this scheme.

- *hierarchical*: a hierarchical key graph enables economical key establishment in the presence of large amounts of subjects. In such a setting, subsets of subjects $\mathcal{V}_{C_i}$ form local cliques and share pairwise keys within each of the subsets. From each of these subsets $\mathcal{V}_{C_i}$, a subject is "elected" to be the representative for their subset $\mathcal{V}_{C_i}$. All representatives form another clique where keys are established amongst all pairs of representatives. Depending on the actual number of subjects in the system, more layers of this hierarchy can be introduced. Herbivore [61] uses a similar idea, but connects the respective subsets $\mathcal{V}_{C_i}$ using Chord [136].

Figure 49 visualizes the aforementioned key graph topologies to illustrate the differences in the complexity of the key establishment and management. Connections between two subjects represent a shared key between the two respective subjects.

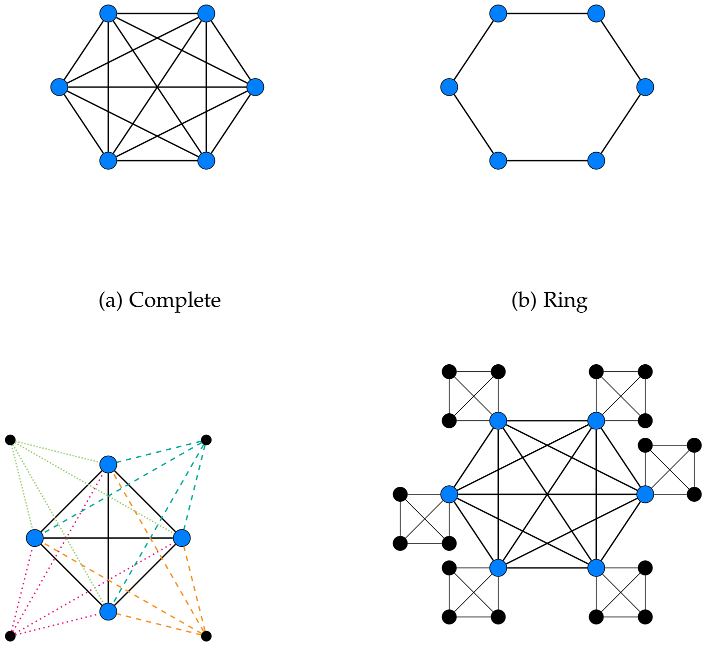
To communicate, each participant v_k creates a message m_{v_k} by combining all their established keys $k_{k,l}$ with either 0 (i.e., null) or the message m, if the subject is the sender $v_k = s$:

$$m_{v_k} = \bigoplus_{v_l \in N_{KG}(v_k)} k_{k,l} \oplus \begin{cases} m, & \text{if } v_k = s \\ 0, & \text{otherwise} \end{cases} \tag{37}$$

Hereby, $N_{KG}(v_k)$ describes the neighbors of v_k in the key graph, i.e., the subjects with which v_k establishes keys for the communication in the DCnet.

6.2.3.2 *Communication in DCnets*

The messages m_{v_k} are then sent to all subjects whose key is being used in the XOR-chain ($\oplus$). By collecting all messages, each of the subjects is able to compute the XOR-function over all received messages to rule out the keys and derive the actual message.



(a) Complete

(b) Ring

(c) Subcliques

(d) Hierarchy

Figure 49: Key Graph Topologies for DCnets

Example 29: Subjects a to d establish their keys in a complete key graph, i.e., subject a has keys $k_{a,b}$, $k_{a,c}$, and $k_{a,d}$—subjects b to d have the keys likewise. Subject a acts in this example as sender s. All subjects compose their messages $m_{\{a,b,c,d\}}$ as follows:

$$\begin{aligned}
m_a &= k_{a,b} \oplus k_{a,c} \oplus k_{a,d} \oplus m \\
m_b &= k_{b,a} \oplus k_{b,c} \oplus k_{b,d} \oplus 0 \\
m_c &= k_{c,a} \oplus k_{c,b} \oplus k_{c,d} \oplus 0 \\
m_d &= k_{d,a} \oplus k_{d,b} \oplus k_{d,c} \oplus 0
\end{aligned}$$

The messages are then exchanged and each participant can derive the message m as follows:

$$\begin{aligned}
m_a \oplus m_b \oplus m_c \oplus m_d = & k_{a,b} \oplus k_{a,c} \oplus k_{a,d} \oplus m \\
& \oplus k_{b,a} \oplus k_{b,c} \oplus k_{b,d} \oplus 0 \\
& \oplus k_{c,a} \oplus k_{c,b} \oplus k_{c,d} \oplus 0 \\
& \oplus k_{d,a} \oplus k_{d,b} \oplus k_{d,c} \oplus 0 \\
= & k_{a,b} \oplus k_{b,a} \oplus k_{a,c} \oplus k_{c,a} \\
& \oplus k_{a,d} \oplus k_{d,a} \oplus k_{b,c} \oplus k_{c,b} \\
& \oplus k_{b,d} \oplus k_{d,b} \oplus k_{c,d} \oplus k_{d,c} \\
& \oplus 0 \oplus 0 \oplus 0 \oplus m \\
= & m
\end{aligned}$$

By this dependence on the XOR-function, the presence of multiple senders s_i and s_j causes a collision and results in a scrambled message. The efficiency can be improved by using a message-vector, where subjects can place their message at different positions to reduce the probability of collisions.

An adversary, however, may decide to jam the communication by replacing their "0" content with random bits, which would yield a collision for each message. To counter this, several authors [22, 30, 63] propose trap-functionalities to identify (but not de-anonymize) the jamming subject; the jamming subject can then be excluded from the system.

6.2.3.3 *Anonymity in DCnets*

In [22], Chaum proves the sender anonymity. For that, Chaum shows by reduction of the applied encryption to one-time pads (see Section 6.2 for the security and anonymity of one-time pads) that the information leakage of (for an adversary $\mathfrak{A}$) visible variables—the messages of the subjects $\mathcal{V} \setminus \mathfrak{C}$ encrypted using the encryption keys on vectors of the prime field $\mathsf{GF}(2)$—is restricted to the parity of the respective vectors. With that reduction, Chaum proves that the remaining set of $\mathcal{V} \setminus \mathfrak{C}$ acts as irreducible anonymity set for the senders $s_i \in \mathcal{S}_t \subseteq \mathcal{V} \setminus \mathfrak{C}$.

6.3 FOCUSSING ON SENDER PROTECTION, ARE RECIPIENTS PROTECTED?

This thesis contributes an approach to protect sender anonymity. This focus raises the question of the protection recipient anonymity. This section outlines why sender anonymity is a particular challenge and why current approaches for recipient protection, for example, Daubert et al.'s *probabilistic forwarding* and *shell game* [37] cannot protect sender anonymity.

In this section, first the information leakage by topological properties are formalized in Section 6.3.1. After that, Daubert et al's recipient protection measures are introduced in Section 6.3.2 and the challenges for sender anonymity are introduced in Section 6.3.3.

6.3.1 *The Issue of Topological Leaks*

Communication overlays O_t leak information about senders and recipients based on their respective positions in the overlay.

The overlay establishment mechanisms introduce this information leakage. As such, either senders s or recipients r initiate the overlay establishment. By this initialization, they create a leaf (or root for senders) in the communication overlay; only senders and recipients can establish these endpoints of an overlay. Other subjects $v \notin \mathcal{S}_t \cup \mathcal{R}_t$ will only join a communication overlay O_t as brokers b if they are required to connect a sender s or recipient r to the communication overlay. Thus, overlay establishment mechanisms place brokers b only at inner positions in the communication overlay O_t but not on endpoints of the communication overlay.

The information leakage is prevalent amongst all communication systems, for example:

- *Pub/Sub-based Communication Overlays*: Recipients r or senders s initiate the overlay establishment. As such, some of them form the endpoints of the established communication overlay; others may be required to relay messages to subsequent recipients and are, therefore, "inner" subjects.

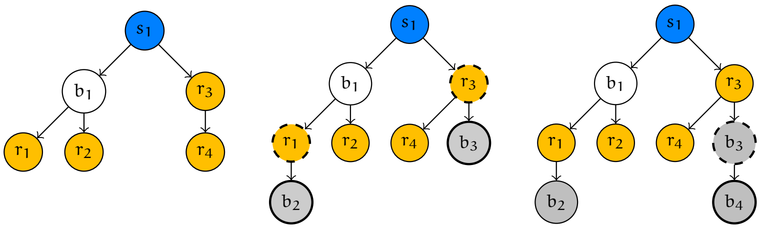
- *Web Mixes & Tor*: Web MIXes [15] and Tor [46] (see Sections 3.3.2 and 3.3.3) establish point-to-point connections between a sender s_i and a recipient r_j. Naturally, both are leaf subjects in their respective communication overlay.

6.3.2 *Obfuscating Recipients due to Topology Adaption*

Daubert et al. [37] propose two mechanisms, namely probabilistic forwarding, and shell game, to adapt the topology of the communication overlay O_t. They aim to reduce the information leakage from recipients in leaf positions. First, additional subjects are opportunistically included in the communication overlay to extend the dissemination paths beyond only leaf-recipients. Second, the topology itself is altered by position swaps between subjects, where two subjects exchange their neighborhoods and positions in the communication overlay O_t.

PROBABILISTIC FORWARDING Using probabilistic forwarding, recipients request other subjects to join the communication overlay O_t opportunistically. For that, brokers b_k and recipients r_j evaluate for each neighboring subject whether to request them to join the communication overlay O_t as additional brokers extending the dissemination paths beyond the original leaves. The opportunistically joining subjects evaluate their neighbors themselves again. Figure 50 visualizes this mechanism: left, the dissemination tree without probabilistic forwarding; the recipients r_1, r_2, and r_4 are exposed as being leaf-subjects. In the middle, recipients r_1 and r_3 are including one of their neighboring subjects each; on the right, broker b_3, which was included by r_3, includes one of its neighbors.

By including more subjects into the communication overlay, the anonymity set increases, and the adversary $\mathfrak{A}$ can no longer identify a leaf as a recipient.

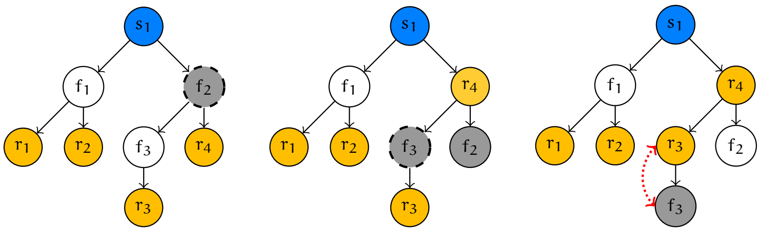


(a) no probab. forw. (b) probab forw. (1). (c) probab forw. (2).

Figure 50: Obfuscating recipients using probabilistic forwarding [37]

SHELL GAME Using the shell game, any two subjects v_l, v_m being part of the communication overlay O_t can swap their positions. For that, they exchange their neighborhood sets $N^{+/-}(v_l)$ and $N^{+/-}(v_m)$. By this exchange, leaf-positioned recipients may rotate into the inner part of the communication overlay O_t—reducing the ratio of recipients in exposed leaf positions.

In the shell game, subject v_l evaluates with a decaying function whether to perform a position swap or not. If subject v_l decides to perform the shell game, v_l chooses a neighbor $v_m \in \{N^+(v_l) \cup N^-(v_l)\}$ and initiate the two-phase commit swapping-protocol where v_l and v_m exchange their neighborhoods and inform their respective neighbors. The decaying function enables the shell game to increase the number of brokers in leaf positions and still reduces the number of position swaps over time. Figure 51 shows three snapshots of a communication overlay O_t with two swaps: first broker b_2 swaps position with recipient s_4; second, broker b_3 swaps with recipient s_3. However, the shell game is not limited to broker–recipient pairs but can be performed with any two subjects.



(a) initial O_t. (b) O_t after shell game #1. (c) O_t after shell game #2.

Figure 51: Randomizing the topology using the Shell Game [37]

resulting recipient protection Both probabilistic forwarding and shell game work together to increase the mixture of recipients and brokers in leaf positions. By bringing brokers into leaf positions—and rising their ratio in the mix of leaf subjects—the information leakage caused by topology is limited.

Daubert et al. [37] use a simulative evaluation to show that the combination of probabilistic forwarding and shell game can easily achieve a mix of recipients and brokers in leaf positions of 0.5. As a result of this, if the adversary $\mathfrak{A}$ picks a random subject in leaf position, their probability of having a recipient is worse than a (fair) coin flip.

6.3.3 *The Challenge with Senders*

Senders s may emit a message at any point in time. Thus, the communication system has to be prepared to hide the sender at every possible point in time.

The lack of knowledge about a sender's message emission leads to two intuitive approaches:

1. message dissemination with opportunistic subjects
2. dummy messages for covering communication messages

protection by opportunistic subjects In the first approach, the communication overlay O_t may be randomized and extended using mechanisms like the aforementioned probabilistic forwarding and shell game. However, both require not only synchronization but also that the sender s can transfer the actual communication message m_t^{com} in a concealed fashion. At first sight, one might think of embedding the message in regularly exchanged heartbeat messages that are a necessity in every P2P system. However, as heartbeat messages are usually by magnitude smaller than a communication message; this restriction of size requires the sender to include an actual communication message in multiple heartbeat messages. This increases the communication delay.

These issues lead to the second, following approach where communication between subjects is hidden in a large number of dummy messages.

DUMMY MESSAGES FOR COVERING SENDERS Dummy messages can cover communication messages, and, thus, senders, by all subjects continuously sending messages in the system all the time. This prevents an adversary $\mathfrak{A}$ from identifying the sender of an actual message. This approach can be performed in two manifestations:

1. Creation of "container" messages that are (randomly) forwarded until they are replaced with a real communication message by a sender s. This requires communication messages m^{com} to be distinguishable from containers from an internal perspective; otherwise, a sender would not be able to select an empty container for replacement and might replace a communication message—leading to a message loss. While being distinguishable, the containers have to be indistinguishable by an adversary $\mathfrak{A}$ to provide cover for senders. The adversary can mask out container messages if communication messages m^{com} and container messages are distinguishable; as a result of this, the adversary can trace communication messages m^{com} without container messages once again.

2. Utilization of all connections using "cover traffic" (see Section 2.4.2.5 for the cover traffic primitive) to hide communication message behind noise. In contrast to 1., cover traffic requires all subjects to utilize all connections at all time. Hence, an encrypted message is hidden as it is not distinguishable from the random noise generated by all subjects.

Both, container messages and cover traffic lead to efficiency issues. For the latter technique, the efficiency issue is analyzed in the following section in more detail.

6.4 COVER TRAFFIC REVISITED

Cover traffic is the state of the art method to protect senders from de-anonymization attempts; all anonymous communication systems [15, 29, 30, 34, 58, 61, 87, 133, 146] of the last decade that provide sender protection use mechanisms that can be reduced to the mechanism of cover traffic.

As introduced in Section 2.4.2.5, cover traffic is based on the assumption that a global adversary $\mathfrak{A}_{p,g}$, for example, the service provider, cannot distinguish an encrypted message from random noise that is utilizing a connection. Therefore, cover traffic requires that all subjects utilize all unused connections with cover messages m^{cov}, i.e., send a cover message m^{cov} on all connections that are not used to relay/send a communication message m^{com}. For all connections, both directions—if existing—have to be considered for cover traffic. Such an implementation of cover traffic does not only provide unlinkability or anonymity—but it also *provides untraceability* against a non-internal adversary $\mathfrak{A}_{\_,g}$. A service provider, for example, cannot recognize a communication when cover traffic is applied in their system.

However, while protecting senders and recipients, cover traffic adds massive load to the communication system as all connections are utilized at all times. Facing heterogenous participants with different capabilities to communicate, the message load of cover traffic yields problems and may lead to an anonymous but impractical communication system.

Considering the transmission of a single communication message m_t^{com} with cover traffic, the utilization with communication messages can be calculated as formalized by Equation (38).

$$\mathrm{util}_{com} = \frac{m_t^{com}}{m^{cov}} = \frac{|\mathcal{E}_t|}{|\mathcal{E}| \cdot \mathrm{diam}(O_t) - |\mathcal{E}_t|} \tag{38}$$

Every connection in the communication overlay $\mathcal{E}_t$ has to be utilized with the communication message once. As messages are relayed every round, the "time" from sender to the farthest recipient is defined by the diameter of the communication overlay $\mathrm{diam}(O_t)$, i.e., by the longest shortest path in the communication overlay. While the message is being transmitted, i.e., during "$\mathrm{diam}(O_t)$-many" rounds, all unused connections have to be utilized with cover messages. Therefore, number of cover messages m^{cov} can be calculated by multiplying all connections in the system ($\mathcal{E}$) times the diameter of the communication overlay $\mathrm{diam}(O_t)$, i.e., "utilizing all connections at all time", minus the number of connections in the communication overlay ($\mathcal{E}_t$) as these are utilized with the communication message m_t^{com} once. Example 30 and Figure 52 exemplify the utilization to show the load induced by cover traffic. The overall utilization decreases with increasing difference between $|\mathcal{E}_t|$ and $|\mathcal{E}|$ and increases with the number of subjects being involved in communications.

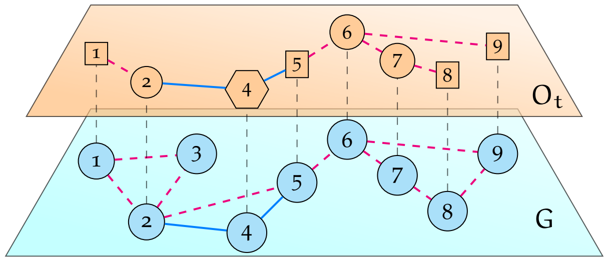


Figure 52: Cover traffic on the example system from Figure 8. Connections colored for the first communication round with cover traffic: blue connections are used to transmit the communication message m_t^{com}, red dashed connections are used to transmit cover messages m^{cov}.

Example 30: Assume a communication system with a structure as depicted in Figure 52. Cover traffic generation considers the connection of the logical underlay G; this is the layer which is visible to the adversary $\mathfrak{A}_{p,g}$ and does not distinguish between different communication overlays O_t.
Sending a single message $m_t^{com}(1)$ from sender $s = 4$ to the recipients $\mathcal{R}_t = \{1, 5, 8, 9\}$ requires 4 rounds ($\mathrm{diam}(O_t) = 4$ between $s = 4$ and $r = 8$). G contains $|\mathcal{E}| = 24$ directed connections.
In the first round, subject 4 sends the message m_t^{com} to subjects 2 and 5. The remaining 22 connections are utilized with cover messages m_{cov}.

round rd	# com. msgs	# cov. msgs	ratio $\frac{com}{cov}$
$rd = 1$	$2\ (4 \to \{2, 5\})$	22	$\frac{1}{11} \approx 0.09091$
$rd = 2$	$2\ (2 \to \{1\}, 5 \to \{6\})$	22	$\frac{1}{11} \approx 0.09091$
$rd = 3$	$2\ (4 \to \{7, 9\})$	22	$\frac{1}{11} \approx 0.09091$
$rd = 4$	$1\ (7 \to \{8\})$	23	$\frac{1}{23} \approx 0.04348$
	$\sum 7$	$\sum 89$	$\frac{7}{89} \approx 0.07865$

The per-round utilization with communication messages in this example is between 4.35%–9.09%; the overall utilization util_{com} is approximately 7.87%.

Research Question 13 *Can the efficiency of cover traffic be improved while the anonymity is preserved at a "reasonable" level?*

6.4.1 *Parameterized Cover Traffic*

The lack of efficiency is caused by the constant utilization of all connections of all subjects which is inherent to the mechanism cover traffic [15, 29, 30, 34, 58, 61, 87, 133, 146]. To reduce the resulting overhead, i.e., to improve utilization and, thus, improve the efficiency of cover traffic, a set of parameters has been derived. These parameters control the aspects of cover traffic that control its efficiency: 1) *subject participation probability*, 2) *connection utilization probability*, and 3) *temporal behavior*.

SUBJECT PARTICIPATION PROBABILITY The subject participation probability (spp) enables the system to produce spotty cover traffic—cover traffic that resembles communication behavior where not every subject sends a message at every point in time. Lowering this probability reduces the number of subjects that generate cover messages m^{cov} and, with that, improves efficiency. Subjects evaluate the subject participation probability spp in every round to balance the load of cover traffic generation among all subjects. Assessing the subject participation probability spp once would free some subjects from the burden of cover traffic while others have the same obligations as before.

Proposition 13 *Controlling the number of subjects participating in the generation of cover traffic* m^{cov} *with the* subject participation probability spp *influences the induced communication overhead.*

CONNECTION UTILIZATION PROBABILITY The connection utilization probability (cup) transfers the concept of adjustable intensities to the connections—after having evaluated their subject participation probability spp, subjects that generate cover traffic decide which of their connections to populate with cover traffic m^{cov}. Subjects evaluate the connection utilization probability cup in every round if they evaluated the subject participation probability spp positively before. Subjects evaluate the connection utilization probability cup regularly to keep the cover traffic load balanced for all subjects.

Proposition 14 *Controlling the utilization of connections that are populated with cover traffic* m^{cov} *with the* connection utilization probability *influences the induced communication overhead, similar to the subject participation probability-based overhead control.*

DYNAMIC CONNECTION UTILIZATION PROBABILITY ADAPTATION When a subject v is a broker $b \in \mathcal{B}_T$ the overlay O_t which has to relay an actual communication message m_t^{com}, subject, the usage of the connection utilization probability cup as is leads to a statistically recognizable distortion. Over time, an adversary $\mathfrak{A}_{p,g}$ may learn to trace the flow of communication messages by tracking these distortions. Thus, the connection utilization probability cup has to be dynamically adapted, whenever the subject has to relay a communication message m_t^{com}; the adaptation follows from Equation (39) where $d^{-}(b)$ denotes the number of connections of subject b and $d_t^{-}(b)$ denotes the respective number of outgoing connections of b that are part of the communication overlay O_t as described in Equation (40). After relaying communication messages m_t^{com}, the adaption is reverted to the simple connection utilizatio probability cup.

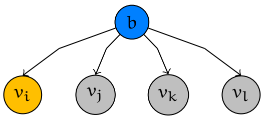
$$cup_{mod} = \frac{cup \cdot d^{-}(b) - d_t^{-}(b)}{d^{-}(b) - d_t^{-}(b)}, \tag{39}$$

$$d_t^{-}(b_l) = |\{e_i : e_i = (b_l, b_k) \in \mathcal{E}_t\}| \tag{40}$$

The adaptation of the connection utilization probability cup to cup_{mod} accounts for the necessary utilization of overlay connections $e_i \in \mathcal{E}_t$ of a subject b and reduces the connection utilization probability for the connections that are *not* part of the communication overlay such that the overall connection utilization probability stays stable even when relaying communication messages. Example 31 visualizes the necessity to adapt the connection utilization probability when a subject b has to relay communication messages m_t^{com}.

If the number of connections in O_t results in a negative modified connection utilization probability cup_{mod}, the distortion cannot be accounted for without delaying the communication messages m_t^{com}.

Example 31: Subjects that are forwarding communication messages m_t^{com} introduce distortions with the generation of cover traffic when using the connection utilization probability cup. Assume the following subject b, which is part of the of the overlay O_t that has four outgoing connections to v_i–v_l, neighbor v_i is also part of the overlay O_t. The connection (b, v_i) is therefore part of the overlay O_t as well.



The adapted connection utilization probability cup_{mod} accounts for the increased probability due to relaying communication messages m_t^{com} by reducing the connection utilization probabilities cup of the remaining connections:

cup	exp. num. of msgs	actual num. of msgs	cup_{mod}	exp. num. of msgs
1.0	4	4	1.0	4
0.75	3	3.25	0.6667	4
0.5	2	2.5	0.3333	2
0.25	1	1.75	0.0	1
0.0	*0*	*1*	*-0.33 (0)*	*1*

Proposition 15 *The adapted connection utilization probability* cup_{mod} *prevents the distortion caused by handling* m_t^{com} *as far as possible.*

TEMPORAL BEHAVIOR Often, it is not required to have cover traffic in *every* possible round ("permanent" or "constant", see Section 2.2.2) as communication takes some time as messages have to be processed before relaying them or replying to them. The emerging overhead can be reduced further by pausing the generation of cover traffic in intervals to use these "idle" times to improve efficiency.

Depending on the communication scenario, the two natural options for the temporal behavior of cover traffic generation are as follows:

- *Intervals:* using interval-based cover traffic, subjects generate cover traffic in configurable sized intervals. This initialization method is suited for, for example, machine-to-machine communication where communication occurs on a regular interval. The size of the interval can be adapted to match the requirements of the specific communication scenarios to avoid prohibitive delays.

- *Request:* using request-based cover traffic, subjects control the generation of cover traffic based on requests. Subjects request the generation of cover traffic, for example, using regularly exchanged heartbeat messages of the underlying P2P system. It is essential to hide the requests to avoid that the adversary $\mathfrak{A}_{p,g}$ learns about the communication—and the sender of communication messages—by tracking the requests. Also, the requests have to be relayed to enable larger groups of subjects to emit cover messages. This setting avoids the generation of cover traffic when no subject intends to emit a message and, thus, reduces the overhead when, for example, people are communicating.

Proposition 16 *Avoiding constant cover traffic improves the efficiency of communication that is protected by cover traffic. Interval-based cover traffic generation protects fixed intervals, for example, to provide anonymity for machine-to-machine communication. Request-based cover traffic generates cover traffic only upon request, for example, to account for irregular communication patterns of people.*

THE NECESSITY OF RELAYING COVER TRAFFIC When changing the temporal behavior of cover traffic generation, subjects have to align their communication behavior with the generation of cover messages m^{cov}. This alignment is important to obfuscate the communication message m_t^{com} in a set of cover messages m^{cov}. However, this obfuscation is not only important for the initial emission of a communication message m_t^{com} at the sending subject but also for the subsequent relaying of the message. Otherwise, an adversary $\mathfrak{A}$ is not able to identify the sender of the communication message but, as the next relaying subject is not obfuscated, can derive the first relaying subject. With the knowledge about the first relaying subject, the sender has to be one of this subject's neighbors—this information leakage reduces the sender anonymity set size from $|\mathcal{V}|$ to number of *common* neighbors of the first relaying subjects of a message $|\{\cap N^{-}(v_l) : v_l \in N_t^{-}(s_i)\}|$.

There are two possibilities to circumvent this information leakage:

- *Delaying* the communication message until the next cover traffic interval is present—or the next cover traffic request is performed.
- *Relaying* cover traffic to simulate ongoing communications instead of putting cover messages only on one connection where the recipients discard them.

While the first possibility would unnecessarily delay communication, the latter solution provides a trade-off between efficiency and anonymity. Upon receiving a cover message m^{cov}, a subject decides whether to relay the cover message or not by using the *relay probability* rp. The relay probability rp is the adaptation of the subject participation probability spp for rounds in which, due to the temporal behavior, no new cover messages m^{cov} are emitted. As such, the relay probability should be configured such that it is in the order of magnitude of the subject participation probability.

The cover messages m^{cov} will decay with the relay probability rp; subjects will terminate the relaying eventually. With that, the number of cover messages m^{cov} decreases over time.

6.4.2 *Quantitative Simulation*

This section discusses the influence of the randomized generation of cover messages on efficiency and anonymity. For that, a simulation study is conducted to compute the efficiency improvements and changes in anonymity.

SIMULATION SETUP Efficiency, as well as anonymity, are in their absolute numbers depending on the underlying network; however, their relative improvement—or deterioration—is independent of the underlying network structure network. This simulation focusses on the evaluation of the influence of the randomized cover traffic initialization; the simulation is conducted only on social networks that are generated following the Barabàsi-Albert model [13].

Following Section 4.4.3, the social networks are generated with $1,000$ subjects, $m_0 = 2$, and $m = 3$; on each network, 1 subject is randomly chosen to be the sender s, a ratio of 0.015 of subjects (i.e., on average 15 subjects) are chosen as recipients $r \in \mathcal{R}_t$. Each experiment is repeated 50 times to reduce the influence of randomly chosen sender and recipients, as well as the influence of neighbor-selection during network generation. Each simulation measures efficiency and anonymity over 20 messages that are emitted by the sender s—each message is given 50 rounds to be relayed from the sender to the recipients.

Efficiency is measured using the *noise to content ratio* ncr which is the number of cover messages m^{cov} which is sent per communication message m_t^{com}; the lower the noise to content ratio ncr, the better the efficiency. Equation (41) shows the calculation of the noise to content ratio ncr which measures the number of cover messages m^{cov} sent per communication messages m_t^{com} and is summed over all rounds.

$$ncr = \frac{|\sum_{rd \in RD}\{m^{cov}\}_{rd}|}{|\sum_{rd \in RD}\{m_t^{com}\}_{rd}|} \tag{41}$$

Anonymity is measured using the number of *remaining cover subjects* rcs, which collects a set of subjects, that emit a message m^{cov} together with the sender. These sets are derived for every round in which the sender emits a message m_t^{com}; then, the intersection of the sets is established. This approach follows an adversary $\mathfrak{A}_{p,g}$ that executes an intersection attack [36, 143]. This adversary $\mathfrak{A}_{p,g}$ resembles, for example, the service provider.

The overlay establishment itself does not influence cover traffic. To minimize start-up time and message delivery time, a conventional overlay based on flooding is used to connect sender and recipients.

Table 24: Simulation Parameters

Parameter	Value
Repetitions	50
Rounds per repetition	50 (overlay establ.) + 1050 (comm.)
Network size ($\|\mathcal{V}\|$)	$\{1,000\}$
Recipients ($\|\mathcal{R}_t\|$)	0.015 of $\mathcal{V}$ (on average 15 subjects)
Network type	social
Network configuration	$m_0 = 2$, $m = 3$
Cover Traffic Configuration:	
Subject Participation Probability	$[0, 1]$ in steps of 0.1
Connection Utilization Probability	$[0, 1]$ in steps of 0.1
Temporal Behavior	{constant, interval(10), request}
Relay Probability	$[0, 1]$ in steps of 0.1

6.4.2.1 *Improving the Efficiency of Cover Traffic*

The reduced amount of overhead of cover traffic with varying (combinations of) subject participation probability spp and connection utilization probability cup after sending 20 messages from the sender $s \in \mathcal{S}_t$ to the recipients $r \in \mathcal{R}_t$ is depicted in Figure 53; Figure 53 visualizes the noise to content ratio ncr as introduced by Equation (41).

The following paragraphs discuss both the influence of the subject participation probability spp and connection utilization probability cup and the—not shown—influence of the temporal behavior and relay probability rp.

VARYING SUBJECT PARTICIPATION AND CONNECTION UTILIZATION Both the subject participation probability (spp) and the connection utilization probability (cup) have the expected linear effect on the noise to content ratio in Figure 53.

- *Varying the subject participation probability:* Lowering the probability will increase the number of subjects that are not generating cover messages m^{cov} on their incident connections. The noise to content ratio is linearly correlated with the number of subjects participating in the generation of cover traffic.

- *Varying the connection utilization probability:* Lowering the probability with which a subject is utilizing incident connections with cover messages m^{cov} is linearly correlated with the noise to content ratio as well. The probability of utilizing a connection is directly correlated with the number of cover messages m^{cov} and therefore influencing the noise to content ratio.

This correlation follows from the formalization of the noise to content ratio ncr in Equation (45) which is based on the following steps. Equation (42) provides the number of rounds with communication messages rd_{com}. $\sum_{rd \in RD} |\{m_t^{com}\}_{rd}|$ sums the number of communication messages m_t^{com} observed. By dividing this sum by the number of connections in O_t, the number of messages emitted by senders $s \in \mathcal{S}_t$ is the result. The diameter of the communication overlay $diam(O_t)$ denotes the number of rounds that is required to transmit a message from a sender $s \in R$ to all recipients $r \in R$.

$$rd_{com} = \frac{\sum_{rd \in RD} |\{m_t^{com}\}_{rd}|}{\mathcal{E}_t} \cdot diam(O_t) \tag{42}$$

The noise to content ratio per communication message $ncr_{m_t^{com}}$ is given by:

$$ncr_{m_t^{com}} = \frac{|\mathcal{E}| \cdot diam(O_t) - |\mathcal{E}_t|}{|\mathcal{E}_t|} \tag{43}$$

With $|\mathcal{E}| = \sum_{v \in \mathcal{V}} d^-(v)$ and $|\mathcal{E}_t| = |\bigcup_{v \in \mathcal{V}_t} N_t^-(v)|$ follows:

$$ncr_{m_t^{com}} = \frac{\sum_{v \in \mathcal{V}} d^-(v) \cdot diam(O_t) - |\bigcup_{v \in \mathcal{V}_t} N_t^-(v)|}{|\bigcup_{v \in \mathcal{V}_t} N_t^-(v)|} \tag{44}$$

In rounds without communication message m^{cov}, the noise to content ratio $ncr_{!m_t^{com}}$ is naturally given by $|\mathcal{E}|$.

Combined with the sequence of observed rounds RD, Equation (45) provides the overall noise to content ratio.

$$\begin{aligned} ncr &= ncr_{m_t^{com}} \cdot \frac{rd_{com}}{diam(O_t)} + ncr_{!m_t^{com}} \cdot (|RD| - rd_{com}) \\ &= \frac{\sum_{v \in \mathcal{V}} d^-(v) \cdot diam(O_t) - |\bigcup_{v \in \mathcal{V}_t} N_t^-(v)|}{|\bigcup_{v \in \mathcal{V}_t} N_t^-(v)|} \cdot \frac{rd_{com}}{diam(O_t)} \\ &\quad + \sum_{v_l \in \mathcal{V}} (d^-(v_l)) \cdot (|RD| - rd_{com}) \end{aligned} \tag{45}$$

The parametrization of spp and cup is then naturally applied by first evaluating the subjects and then their respective connections

$$\sum_{v \in \mathcal{V}} d^-(v) \rightarrow \sum_{v \in \mathcal{V}} d^-(v) \cdot spp \cdot cup \tag{46}$$

Combining the parameterized participation with the noise to content ratio ncr in Equation (45) explicates the visually recognized linear correlation of the noise to content ratio ncr with each of the probabilities spp and cup.

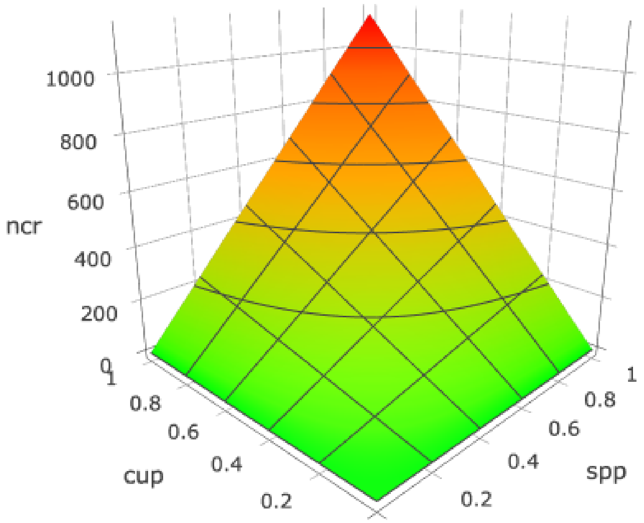


Figure 53: Noise to content ratio ncr with varied subject participation probability spp and varied connection utilization probability cup.

VARYING TEMPORAL BEHAVIOR Until now, cover messages m^{cov} are generated in every round; the efficiency improvement is based on the reduced and randomized subject participation and connection utilization.

Adapting the temporal behavior yields another stage of efficiency improvement. In both the interval-based and the request-based configuration, the relay probability rp expands the subject participation into rounds without generation of new cover messages m^{cov}.

The generation of new cover messages m^{cov} is controlled with varying subject participation probability spp and connection utilization probability cup. When subjects generate cover traffic, i.e., when the interval-based generation is due, or cover traffic is requested, subjects in the system generate on average the following number of cover messages m^{cov}:

$$\overline{|m^{cov}|} = |\mathcal{V}| \cdot \mathrm{spp} \cdot \mathrm{cup} \tag{47}$$

In the rounds in between, each subject that receives a cover message m^{cov} decides whether or not to relay the message using the relay probability rp. That implies the number of cover messages is decreasing with each additional round. The effect is linear with rp (and cup).

The efficiency improvement is larger as with cover traffic with constant generation. The factor of improvement depends on the actual temporal behavior and the relay probability $\mathtt{rp}$; usually, it follows the same progression as visualized in Figure 53.

6.4.2.2 *Cover Traffic in the Tension of Efficiency and Anonymity*

The protection by cover traffic bases on the obfuscation of communication messages m_t^{com} in cover messages m^{cov}. Reducing the number of cover messages may, therefore, have a negative impact on anonymity. Figure 54 depicts the number of remaining cover nodes $\mathtt{rcn}$ after transmitting 20 messages from the sender s to the recipients $r \in \mathcal{R}_t$.

The shown remaining cover subjects $\mathtt{rcs}_s$ are computed as described in Equation (48) by intersecting the active subjects. Active are those subjects, that send at least one message in a window of ± 1 around the rounds $\mathtt{rd}_s \in \mathtt{RD}_{send} \subseteq \mathtt{RD}$ in which the sender $s \in \mathcal{S}_t$ emits a communication message:

$$\begin{aligned} \mathtt{rcs}_s(\mathtt{rd}_s) = |\{v\colon\ & v.\mathtt{sentIn}(\mathtt{rd}_s - 1) \\ & \vee v.\mathtt{sentIn}(\mathtt{rd}_s) \vee v.\mathtt{sentIn}(\mathtt{rd}_s + 1)\}| \end{aligned} \tag{48}$$

Figure 54 visualizes the remaining cover subjects $\mathtt{rcs}$, on the z-axis, dependent of varied subject participation probabilities $\mathtt{spp}$, on the y-axis, and varied connection utilization probability $\mathtt{cup}$, on the x-axis. It shows that a reduction of the probabilities for subject participation and connection utilization reduces the protection of the sender for values of $\mathtt{spp} < 0.5$ and $\mathtt{cup} < 0.4$—larger probability values establish a stable plateau of cover subjects for the sender. Comparing the plateau with the efficiency improvement shown in Section 6.4.2.1, the results promise improved efficiency with stable anonymity through stable numbers of cover subjects.

However, when analyzing the set of cover subjects over time, the anonymity is constantly decreasing. Figure 55 depicts the size of the respective sets of cover subjects for varying settings of the subject participation probability $\mathtt{spp}$ with a fixed connection utilization probability $\mathtt{cup} = 0.4$. Figure 55 visualizes the performance of an adversary $\mathfrak{A}_{p,g}$ who can perform a global traffic analysis attack, i.e., shows the performance of an adversary $\mathfrak{A}_{p,g}$ being able to analyze the message flow in the overall system. By intersecting the sets of cover messages for individual communication messages $\mathtt{rcs}_s$, as given by Equation (49), the adversary

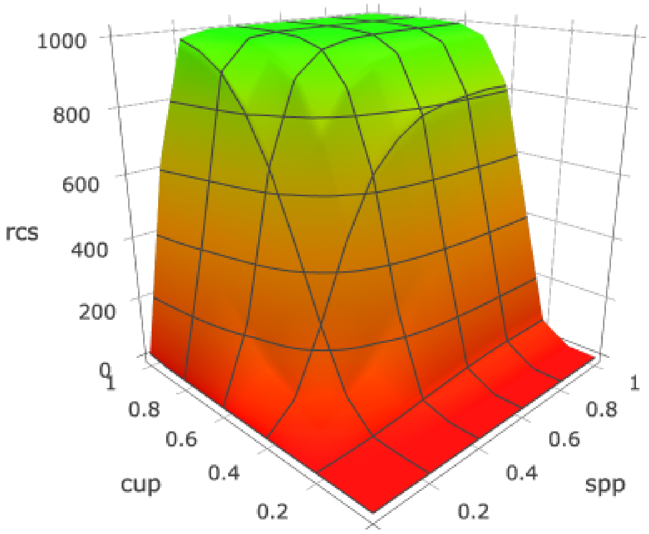


Figure 54: Remaining cover subjects with varied subject participation probability spp and varied connection utilization probability cup.

$\mathfrak{A}_{p,g}$ can rule out cover subjects over time; thus, the adversary can identify the sender s eventually.

$$\begin{aligned} rcs = | \bigcap_{rd_s \in RD_{send}} & \{v_l \colon v_l.sentIn(rd_s - 1) \\ & \vee v_l.sentIn(rd_s) \vee v_l.sentIn(rd_s + 1)\}| \end{aligned} \tag{49}$$

By this intersection attack, the adversary $\mathfrak{A}_{p,g}$ can remove of cover subjects over time. If the system significantly improves efficiency, i.e., configures the spp with a value of around 0.5, the adversary $\mathfrak{A}_{p,g}$ is able to remove 25% of the cover subjects. Over a series of 20 communication messages m_t^{com}, the adversary $\mathfrak{A}_{p,g}$ is able to remove cover subjects; the loss of anonymity is negligible only if the subject participation probability spp is larger than 0.8. Figure 55 also shows the influence of additional changes in the subject participation probability spp.

Additional results with varied connection utilization probability cup are reported in Appendix D.

6.4.3 *Inability of Cover Traffic to Persistently Protect Senders*

The results of Sections 6.4.2.1 and 6.4.2.2 show that the efficiency of cover traffic can be improved by randomized initializa-

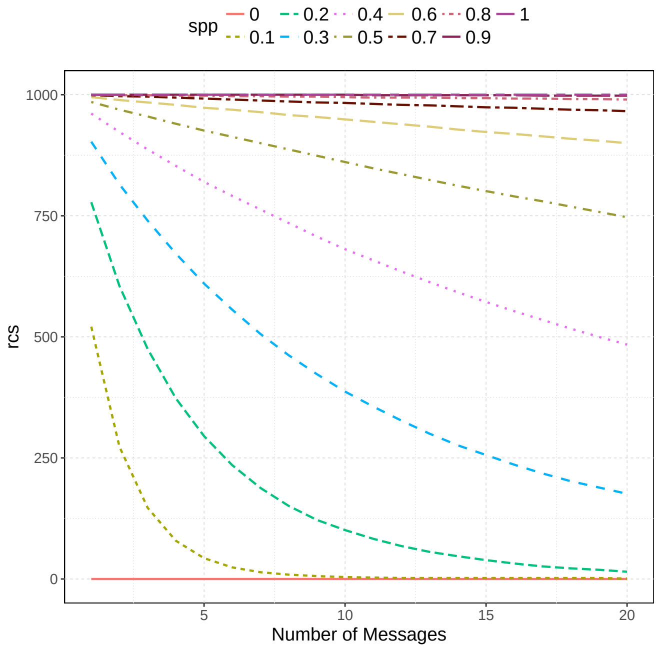


Figure 55: Remaining cover subjects over up to 20 communication messages

tion. However, the results also show that cover traffic with randomized initialization cannot protect sender anonymity over the course of communication—the longer communication lasts, i.e., the more communication messages are exchanged, the higher the chances of an adversary $\mathfrak{A}$ to identify the sender of these communication messages.

The variation of the subjects that generate cover messages $\mathfrak{m}^{cov}$ is the weakness of the randomized cover traffic initialization. However, fixing the set of of cover subjects introduces another set of challenges:

- How are brokers $\mathfrak{b}$ protected? *If brokers* $\mathfrak{b}$ *are not protected, an adversary* $\mathfrak{A}_{p,g}$ *can identify the source by identifying the communication message that is forwarded.*

- How are cover subjects and senders as well as brokers connected? *If the dissemination trees are disconnected—or implausible—, i.e., if the communication overlay looks partitioned to the adversary* $\mathfrak{A}$*, the sender is again unprotected.*

The following section proposes a mechanism that resolves these issues together with efficiency and anonymity considerations to provide efficient and effective sender protection that is independent of the duration of—or number of communication messages in—the communication.

6.5 ADCNETS FOR CONFIGURABLE, EFFICIENT COVER TRAFFIC

This section introduces and evaluates a novel ADCnet-based approach to protect the anonymity of senders using a novel concept for generating secure cover traffic that is not susceptible to traffic analysis or cryptanalysis. Moreover, the presented mechanism provides this cover traffic with provable guarantees and configurable anonymity sets.

In this section, first the concept of *asymmetric* DCnets is introduced in Section 6.5.1, also related approaches are discussed. Section 6.5.2, the adapted and novel ADCnets are established with their three stages: i) initialization, ii) group formation, and iii) message initialization. The introduction is accompanied with recommendations for different variations, depending on the desired properties; these variations have, however, no influence on anonymity-related properties. Lastly, Section 6.5.3 evaluates the novel ADCnet-based approach with a qualitative discussion of the properties of the proposed ADCnets.

6.5.1 *Concept of Asymmetric DCnets*

Symmetric DCnets require every participant of the used key graph to contribute to the communication. Usually, this communication is performed in a broadcast-like manner. This type of communication put an immense load on the system and every subject that participates in the communication.

Asymmetric DCnets (ADCnets) reduce the load of communication by avoiding either large groups or broadcast communication. The following systems from related work use asymmetric approaches

- Herbivore [61] forms small DCnets of dynamic size. Within these small DCnets, subjects communicate using the traditional DCnet technique. Inter-DCnet communication is established by connecting them using a Chord-based network [136].
- Dissent [30] uses a hybrid approach to reduce the message load for subjects. For that, subjects connect to a set

of dedicated entities. These dedicated entities form a symmetric DCnet which anonymizes the communication on behalf of the subjects. This asymmetry requires the broadcast communication only from the entities that form the DCnet; usual subjects need to establish keys with all of the DCnet-entities and send their message to all of them.

The anonymity set is formed by all subjects that are communicating using the system. However, their dynamic behavior, i.e., churn, will reduce anonymity over time, similar to the evaluation of cover traffic shown by Section 6.4.2.2.

- Asymmetric-DCnets [19] are first introduced as a combination of DCnets and additive homomorphic encryption for scenarios like casting votes and privacy-preserving smart-metering applications.

  Subjects form subgroups that establish keys within their groups. Subjects send their message then only to a "tallyman" which can evaluate the messages without being able to contribute parts of the result to specific subjects.

The following ADCnet-based approach is based on the idea of de Borges et al. [19]. The ADCnet-based approach of this thesis goes beyond the concept of de Borges et al. with a novel connection to the ACO-based communication overlay. With this connection, the efficiency of sender protection can be further optimized. Also, this thesis contributes mechanisms to initialize and assemble the group of subject that establishes an ADCnet.

Figure 56

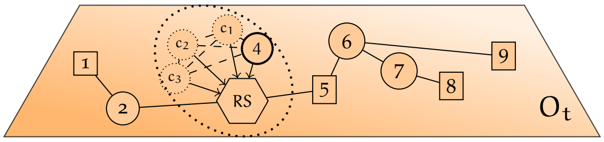


Figure 56: ADCnet-based sender protection: an ADCnet protect the sender 4 in a group of subjects c_1–c_3. The rendezvous subject RS connects acts as "sender" and connects the ADCnet to the communication overlay.

The following sections assume different adversary manifestations ranging from a global passive adversary $\mathfrak{A}_{p,g}$ to local or colluding active adversaries $\mathfrak{A}_{a,l}$ and $\mathfrak{A}_{a,c}$. The adversary manifestation is chosen based on the actual goal of discussion such that the strongest realistic adversary is assumed.

6.5.2 *Introducing ADCnet for Sender Protection*

In this section, the concept of ADCnets for sender protection is described. First, an overview is provided; then, the three phases of the mechanism are described in more detail.

CONCEPT OF ADCNETS FOR SENDER PROTECTION Using ADCnet groups, each sender $s \in \mathcal{S}_t$ hides in a local[1] group of subjects that forms a "small" (and adapted) DCnet. As the participation (or non-participation) of subjects in an ADCnet may leak information about locations of senders s, every subject will join—or establish—an ADCnet eventually.

Within each of these ADCnets, the key graph is a complete graph, i.e., the subjects in a group have pairwise keys established, and the communication graph forms a *star* with a *rendezvous subject* establishing the center of such an ADCnet. This rendezvous subject provides a connection to the remaining communication system.

Depending on the computational resources of a subject, each subject may join multiple ADCnets.

The size of the local ADCnets constrains efficiency and anonymity. Larger groups require more messages—and more cryptographic operations—than smaller ones and have a lower resulting efficiency. However, larger groups also establish larger anonymity sets and therefore higher anonymity. The concept of ADCnets for sender protection is equipped with configurable sized anonymization groups to suit different application scenarios, ranging from low sensitivity to high sensitivity communications.

The anonymity of an ADCnet is grounded on the anonymity of DCnets. As a result of this, the strength of anonymity relies on the number of honest subjects in the group, i.e., the number of subjects $|\mathcal{V} \setminus \mathfrak{C}|$ that are not controlled by the adversary $\mathfrak{A}$.

6.5.2.1 *Initialization*

ADCnets need to be initialized before a subject v intends to send a message with respect to any topic $t \in T$, i.e., before a subject

[1] Local is defined according to the actual application and may lead to, for example, a spatial locality definition or a community or friend-based locality definition

v can take an active role as sender $s \in \mathcal{S}_t$. For that initialization, two intuitive points in time are possible:

- *On-demand* initialization where this subject v will assemble the ADCnet as soon as they decide to enter the active role of the sender s. In order to enter this role, the ADCnet has to be in place.

- *In-advance* initialization requires that every subject v checks at random points in time whether they are part of an ADCnet or not. If they are not part of an ADCnet, they may initiate the assembly of one with a specifiable probability parameter p_{init}.

On-demand initialization improves efficiency as the key establishment and storage as well as the regular communication are only performed if communication is taking place. As subjects set up an ADCnet if they are about to take the sender role, the setup—and existence—of an ADCnet leaks information about the location of senders.

In-advance initialization, however, improves the anonymity. As the ADCnets are established beforehand, the establishment itself does not leak information about the sender s—or the subject v that intends to take that active role. The continuous set up of ADCnets that may not even be used degrades efficiency.

6.5.2.2 *Group Formation*

The concept of ADCnets is a general technique that applies to a variety of applications. Group formation is one of the core elements of this technique. However, ADCnets are independent of the actual group formation mechanism as long as it meets the following requirements:

1. The group size is configurable, and subjects can evaluate whether the configured group size is met by an ADCnet or not—without exposing information about the group participants.

   Subjects need to decide whether enough subjects are already collected to form the group to decide on their own participation. For that, they need to be able to assess the number of subjects and the desired group size. Exposing the subjects that agreed to participate may leak information to an adversary $\mathfrak{A}$ and has to be avoided.

2. Subjects assembled in an ADCnet are similar concerning a definable metric, for example, connectivity or location.

   Subjects that form the ADCnet have to communicate with each other; therefore, subjects should be close, for example, to avoid unnecessary load and delay, or have similar communication capabilities to prevent bottlenecks—and the overloading of some subjects that are less capable.

3. Within each ADCnet, and also outside of the groups, information on whether the group formation was induced by an intent to take an active role as a sender is not leaked.

   If an adversary may learn that a group formation is induced by the intent to take a sender role, it is likely that one of the already collected subjects is an actual sender. The anonymity set of this subject is, therefore, smaller as expected.

4. Within each ADCnet, a random subject is chosen to be the rendezvous subject. This subject is representing the ADCnet to the remaining communication system.

   A random *subject is chosen as rendezvous subject to avoid information leakage. Moreover, if the adversary is able to take over the rendezvous subject deliberately, they can disrupt communication more stealthy by, for example, preventing the connection to the communication overlay. Such an approach is harder to detect in contrast to disrupting the communication by causing collisions (see Section 6.2.3).*

5. An adversary $\mathfrak{A}$ is prevented from taking over targeted ADCnets, e.g., by appropriate (peer-verified) randomization of subject selection for the group formation.

   The anonymity in DCnets—and therefore in ADCnets—relies on the honesty of some *participants, i.e., the anonymity is grounded on the assumption that an adversary cannot take control of all participants in a DCnet. If the adversary $\mathfrak{A}$ could take over a complete DCnet, except the one targeted subject, the adversary $\mathfrak{A}$ can evidently de-anonymize the subject.*

Any mechanism that fulfills the requirements is suitable for the assembly of subjects for an ADCnet.

In the following, two appropriate schemes, the second one adapted from related work, are described in short.

RANDOMIZED GROUP FORMATION Using a randomized approach, subjects assemble a set of participants to establish an ADCnet without involving additional external entities like a trusted third party.

1. An initiator establishes an ID set by adding their own ID and adding one or multiple IDs of known subjects. By adding those IDs, the initiator can hide the origin of the initiation.

   The initiator adds a random number of known subjects to the collection message. If an adversary receives such an initiate-message, they cannot decide easily whether all subjects added themselves (see step 2) or (one of) their predecessor(s) added multiple subjects to add noise. The initiator has to set the target size of the ADCnet to account for the cover-subjects; otherwise, the resulting ADCnet will be too small.

2. The ID-set is forwarded to a random neighbor who will add their own ID. If the configured target-size is not yet reached, the ID-set is forwarded to the next random neighbor.

   Forwarding the ID-set to a random neighbor limits the probability that an adversary $\mathfrak{A}_{\_,l}$ or $\mathfrak{A}_{\_,c}$ will receive this message. It has to be assumed that the adversary $\mathfrak{A}_{\_,l}$ or $\mathfrak{A}_{\_,c}$ fills the remaining places in the ADCnet with subjects under their control, i.e., that the adversary $\mathfrak{A}_{\_,l}$ or $\mathfrak{A}_{\_,c}$ adds subject $c \in \mathfrak{C}$. With this, the adversary could reduce the anonymity set of actual senders in the ADCnet.

3. After collecting a sufficient number of IDs, subjects reconstruct the full set of collected subjects. Subjects that have been added to cover the initialization ignore the establishment of this ADCnet.

   a) Send the collection-message back to its source by matching the neighbors with the IDs in the message. The IDs are now encrypted using a system-wide key to hide the IDs from an adversary $\mathfrak{A}_{p,g}$ that is not a participant of the system. Then, the encrypted IDs are added to a set.

      Every subject relays the collection-message back towards its source. For that, subjects memorize the neighbor from whom they received the collection-message in the first place.

   b) The initiator receives the set containing all encrypted IDs and informs each subject contained about other participants of the ADCnet.

      The initiator can recognize its own collection-message by storing some ID for the message. By also memorizing the intentionally added cover subjects, the initiator can retrieve the actual set of subjects that agreed to participate in the ADCnet.

4. The subjects establish pairwise keys for multiple rounds. For example by using out-of-band communication, or carrying out a key establishment or agreement protocol such as Diffie-Hellmann key exchange [45, 103] or a key derivation function like HKDF [85].

   Pairwise keys are required to use the DCnet-based communication and anonymization. As the anonymization is based on one-time pads, every round a new key has to be used.

5. One of the subjects is selected to be the *rendezvous subject*. This can be achieved by having the subjects draw a random number. Each subject commits their respective number without revealing the number—after all subjects performed their commit, the numbers are revealed to select the rendezvous subject. The commit procedure prevents that an adversary $\mathfrak{A}$ from changing their number to manipulate the selection process.

   The rendezvous subject connects the ADCnet with the communication overlay and acts as a bridge. As such, the rendezvous subject pretends to be the sender(s) s *that are covered in the ADCnet. The randomization balances the load of handling the cryptographic operations of the ADCnet, and the load of handling the communication on behalf of the sender amongst all subjects.*

GLOBAL ADCNET ASSIGNMENT An alternative group formation process can be adapted from Herbivore [61]. Using this process, subjects have to follow an entry protocol with a global ADCnet assignment party.

The global ADCnet assignment party ensures that subjects

- get a random ADCnet assigned. With this, it is ensured that an adversary $\mathfrak{A}$ cannot target a specific ADCnet and take over this ADCnet to de-anonymize a subject.

- obey a rate limit of joins. With this, an adversary $\mathfrak{A}$ is hampered in the whitewashing of their controlled subjects, i.e., removing history of events from maliciously behaving subjects

6.5.2.3 *Communication*

ADCnets improve the efficiency of communication with sender protection by having the subjects in within each ADCnets sending their composed messages only to the respective rendezvous subject instead of circling the message amongst all other participants of their ADCnet.

The rendezvous subject of an ADCnet represents the respective, included senders s. For that, the rendezvous subject acts on behalf of the included senders—without knowing their identities—and follows the overlay establishment protocol, for example, the ACO-based overlay establishment from Chapter 4, to connect the ADCnet to the communication overlay O_t.

For the actual communication, each subject v_k in an ADCnet follows the following steps:

1. select/derive the session keys $sk_{k,l}^{rd}$ for each of other participants v_l (similar to Section 6.2.3, $sk_{k,l}^{rd}$ denotes the key shared between subjects v_k and v_l) for the next communication rd.

2. compose the ADCnet message as introduced in Equation (37) and as follows
   - sender $s = v_k$: $m_s = \bigoplus_{v_l \in N^{\cdot}(s)} sk_{k,l}^{rd} \oplus m_t^{com}$
   - others v_k: $m_{v_k} = \bigoplus_{v_l \in N^{\cdot}(v_k)} sk_{k,l}^{rd} \oplus 0$

3. send the ADCnet message m_{v_k} to the rendezvous subject.

The rendezvous subject removes the session keys and derives the communication message m_t^{com} by combining the received messages using the XOR-function[2] (see Example 29). Then, the rendezvous subject forwards the derived message m_t^{com} to the communication overlay O_t.

In order to be able to communicate, all subjects in an ADCnet have to contribute in in the "correct" rounds; otherwise, the session keys $sk_{k,l}^{rd}$ do not match and an encryption is not possible.

[2] This application of the XOR-function removes only the *ADCnet* encryption but not the end-to-end encryption that ensures confidentiality.

For that, they have synchronize and agree on time-intervals that are used as representation of the concept of rounds (see also Section 2.2.2).

6.5.3 *Qualitative Discussion*

This section evaluates the proposed approach of ADCnets for efficient and effective sender protection. Efficiency related properties are discussed first to understand the impact of ADCnet-based sender anonymization on the communication system. After that, the discussion is revisitedfrom the perspective on effectivity in the provision of sender protection.

This section concludes with a discussion in brief of issues that are inherited from classical DCnets.

6.5.3.1 *Efficiency*

The efficiency discussion consists of two parts: first, the efficiency concerning the actual communication is discussed. Second, the efficiency of group formation and the key establishment is briefly discussed.

COMMUNICATION OVERHEAD With k_{ADCnet} being the number of subjects in each of the ADCnets, each communication message m_t^{com} of a sender s_i in such an ADCnet is accompanied with $k_{\text{ADCnet}} - 2$ messages: every usual subject in the ADCnet sends a message only composed of the respective session keys towards the rendezvous subject, the sender includes the communication message in the message towards the rendezvous subject, and the rendezvous subject does not send an additional message. As a result of this, the noise to content ratio ncr (see Section 6.4.2.1) is as follows:

$$ncr = \frac{k_{ADCnet} - 2}{1} \tag{50}$$

Obviously, the noise to content ratio ncr of ADCnets is linearly dependent on the size of the ADCnets k_{ADCnet}—for n added subjects, n additional messages must be send.

GROUP FORMATION During the group formation, the efficiency depends on the utilized mechanism. Considering the example of gossiping-based randomized group formation (see Section 6.5.2.2), the efficiency is dominated by steps 2 and 3 in which $3 \cdot (k_{\text{ADCnet}} - 1)$ messages are sent:

- $k_{\text{ADCnet}} - 1$ messages to collect the subjects,
- $k_{\text{ADCnet}} - 1$ to reconstruct the set of subjects, and
- $k_{\text{ADCnet}} - 1$ to inform all subjects about the other subjects in the ADCnet.

This shows, the group formation is dependent on the group size—the influence on the efficiency of this dependency hinges on the applied group formation technique.

KEY ESTABLISHMENT The key establishment phase ensures that all subjects within a single ADCnet have their keys shared such that either the session keys are already established (pre-shared keys) or derivable using a key derivation function (using a pre-shared secret as input).

While pre-shared session keys do not influence the efficiency in the key establishment phase for ADCnets, this method is questionable in the application scenario of this thesis where subjects are assumed to be dynamic, and the groups are assembled dynamically.

A commonly accepted mechanism to establish a shared key using an insecure channel is the Diffie-Hellman key exchange protocol [20, 45, 103]. Using the Diffie-Hellman protocol in its authenticated version [20], subjects establish a shared key with three sent messages.

In the key establishment phase of ADCnets of size k_{ADCnet}, all participants have to establish pairwise keys, in sum, $0.5 \cdot k_{\text{ADCnet}} \cdot (k_{\text{ADCnet}} - 1)$ keys have to be established.

The established keys can then be used as input for a key derivation function, for example, HKDF [85], to derive the actual session keys. The derivation of session keys does not require additional communication—yet, the security degrades over time with the number of derived session keys. The number of session keys that are safely derivable depends on the bit-length of the output of the used hash function (in HKDF); after this threshold, a new set of keys has to be established.

As a result of this, the efficiency of key establishment in ADCnets, using the simple (authenticated) Diffie-Hellman protocol, depends with a quadratic factor ("$O(n^2)$") on the size k_{ADCnet} of the ADCnets.

SUMMARY The efficiency of sender protection using ADCnets depends on the size of the actual ADCnets. In stable environments, the costly establishment of ADCnets not performed frequently; the higher costs to form groups and key establishment process occur only occasionally.

The communication overhead itself is linearly dependent on the ADCnet size k_{ADCnet}; only when k_{ADCnet} converges to $|\mathcal{V}|$, the efficiency suffers similar to the cover traffic-based sender protection.

Assume a social network as used in the experiment in Section 4.4.1 with the following configuration:

- $|\mathcal{V}| = 2,000$, $|\mathcal{E}| = 5,994$
- $|\mathcal{E}_t| = 62$, $\mathtt{diam}(O_t) = 5$

Table 25 reports the number of messages that is handled by the communication system during the transmission of one communication message m_t^{com}. For the ADCnets, the number of messages are differentiated for the communication and the establishment phases. The noise to content ratio ncr is calculated for the respective communication phases. ADCnets reveal a higher efficiency, especially considering that the additional load of the group formation and key establishment phase.

Table 25: Comparing the Efficiency of ADCnets and Cover Traffic

Mechanism	Number of Messages	ncr
CT	29,970	482.3871
CT ($spp = 0.5$, $cup = 0.5$)	≈10,000	160.2903

	message	Comm.	Establ.	ncr
ADCnet$_{20}$	m^1	80	57	18
ADCnet$_{20}$	$m^{\geqslant 2}$	80	-	18
ADCnet$_{100}$	m^1	160	297	98
ADCnet$_{100}$	$m^{\geqslant 2}$	160	-	98

6.5.3.2 *Anonymity Protection*

The sender anonymity of ADCnets is provided similar to classical DCnets [22] by reducing them to the anonymity of one-time pads [131, 132].

In an ADCnet, each subject composes its message as by computing the XOR-function with the respective session keys as input (see Equation (37) on page 172). By that construction and given sufficient randomness in each key $k_{k,l}$, the XOR-composition of the session keys creates a uniformly random one-time pad $P = GF(2)^n$ such that the adding of the message $M = P \oplus \{m_t^{com}$ or $0\}$ is still uniformly random and does not yield an information leakage that would affect the sender's anonymity.

$\mathfrak{A}$ controlling an internal subject Obviously, an adversary $\mathfrak{A}_{\_,c}$ knows the session keys that belong to their colluding subjects $\mathfrak{C}$. As a result of this, the anonymity of a sender can be quantified with the anonymity set size $k_{ADCnet} - |\mathfrak{C}|$. Evidently, the sender anonymity can be improved further by establishing larger ADCnets.

$\mathfrak{A}$ controlling the rendezvous subject The rendezvous subject is special in its function as bridge to the communication overlay O_t and the communication system itself. Besides that particular functionality, the rendezvous subjects does not hold any additional knowledge. As a result of this, the rendezvous subject can derive the contained message but cannot identify which of the subjects included the message. The adversary $\mathfrak{A}_{\_,c}$ can remove the keys from its colluding subjects $\mathfrak{C}$. However, the remaining subjects still act as a single anonymity set of size $k_{ADCnet} - |\mathfrak{C}|$.

$\mathfrak{A}$ controlling an outside subject An adversary $\mathfrak{A}$ outside of the ADCnet is not aware of the members of the ADCnet and does not posses the keys necessary to remove members of the ADCnet from the anonymity set.

6.6 Conclusion

This chapter reasoned about the special relevance of sender protection and why sender protection can hardly be achieved with conventional measures like overlay obfuscation; after the reasoning about the importance of sender protection, this chapter analyzed the current state of the art mechanism for sender protection *cover traffic* and a novel proposal, the *ADCnets*.

The reason for the special relevance of sender protection is located in topological information leaks and the unpredictability of the sender behavior. Senders are in particular topological positions in a communication overlay, as are recipients. Senders form the roots of their respective dissemination trees, recipients may form the leaves of the dissemination trees. These positions are caused by the nature of communication systems: senders create or emit information, so they are information sources; recipients consume information, so they are information sinks. Daubert et al. tackled the information leakage for recipients and proposed *randomized forwarding* and the *shell game* to obfuscate the location of recipients by altering the topology. These procedures, however, are limited to recipients and not applicable to provide sender protection.

Moreover, it is not clear upfront at which points in time a sending subject s is going to emit a message. Therefore, sender protection mechanisms have to be in place continuously to protect the sender s. The current state of the art mechanism for sender anonymity is cover traffic; often, cover traffic is disguised by enforcing empty requests (for example, [29]) or dummy requests and messages when no communication takes place (for example, [15, 34, 58, 133, 146]). Cover traffic hides the communication itself in noise, for example, cover messages that appear random and are, thus, not distinguishable from encrypted messages. To enable effective protection, cover traffic needs to be in place continuously. While this continuous utilization allows anonymity for senders *and* recipients, the efficiency degrades massively.

The efficiency can be improved by reducing the number of subjects that generate cover messages and by reducing the number of connections that are populated with these cover messages. Further efficiency improvements can be achieved by altering the temporal behavior when generating cover traffic, i.e., creating cover message only in larger intervals or even only upon request. Implementing these improvements in a randomized fashion to distribute the load over all subjects degrades anonymity if the subject participation probability spp is configured to be below 0.4. The same holds for the connection utilization probability cup. Analyzing the information leakage over time reveals that the adversary $\mathfrak{A}_{p,g}$ can remove the noise over time, i.e., the adversary $\mathfrak{A}_{p,g}$ may identify the senders eventually.

In the second part of this chapter, this thesis proposes the novel mechanism ADCnets to protect senders from de-anonymization and subsequent identification by an adversary $\mathfrak{A}$. Based on the classical anonymization primitive DCnet, senders are hidden in a local group that forms an ADCnet. This group uses a rendezvous subject to enable the overlay establishment on behalf of the actual senders that are included in the ADCnet, and to relay messages from within the ADCnet to the communication overlay O_t. The ADCnet provides provable anonymity against adversaries that are inside the ADCnet, under the assumption that at least one other subject in the ADCnet is not colluding with the adversary; moreover, ADCnets also protect against a global adversary who overlooks the complete message flow within such an ADCnet.

The costs establishment of ADCnets for forming the group and establishing the encryption keys are *quadratic* in the size of the established groups; the establishment of ADCnets is, however, only performed occasionally. The communication *overhead is linear* in the size of the ADCnet. The costs of both the establishment of the ADCnet and the communication depend on the size of the ADCnet, as is the anonymity but inversely proportional. As a result of this, ADCnets provide a configurable trade-off between efficiency and anonymity.

The key lessons of this chapter are as follows:

- Sender anonymity cannot be achieved with overlay adaptation alone and requires continuous protection. Typical mechanisms to provide recipient anonymity are not applicable.

- Cover traffic provides anonymity by hiding communication among noise—communication is untraceable. Anonymity is provided at the cost of drastically reduced efficiency.

- Randomized cover traffic can mitigate the efficiency loss; the efficiency improvement of randomized cover traffic is linear with both the subject participation probability spp and the connection utilization probability cup. However, efficiency improvement also enables intersection attacks; enabling an adversary $\mathfrak{A}$ to reduce anonymity over time and to de-anonymize senders eventually. Improving the efficiency by 50%, the loss of anonymity is 10% after only 20 messages.

- ADCnets provide fixed groups of subjects that provide cover for senders. The cover of ADCnets is computationally secure and cannot be removed by an adversary $\mathfrak{A}$.
- ADCnets enable to trade efficiency and anonymity by being adaptable in their size. Smaller groups favor efficiency, larger groups favor anonymity.
- In ADCnets, the communication overhead increases linear with the size of the respective ADCnets. The group formation is also linearly dependent on the size of ADCnet; the key establishment is quadratic. Both the group formation and the key exchange are only performed occasionally.

7 CONCLUSION

Over the course of this thesis, a novel overlay establishment mechanism was developed. This overlay establishment mechanism enables the assembly of groups that are using communication resources more efficiently than conventional overlay by reducing the load in the system. The pure overlay establishment mechanism was complemented with measures to cope with disruptions by subject churn and provides higher robustness than the state of the art. Additionally, a new efficient scheme for sender anonymity protection that provides computationally secure anonymity for senders was suggested.

This chapter summarizes the core contributions and findings of this dissertation, draws conclusions and presents an outlook to future work.

7.1 SUMMARY AND CONCLUSIONS

(Digital) communication is still becoming ubiquitous and pervaded everyone's life. Communication services are no longer bound to specialized devices like personal computers or mobile phones but are available on devices of all sorts like, for example, smartwatches, smartphones, sensors in smart environments. The congregation of ubiquity and rising heterogeneity with centralized and quasi-monopolistic service providers raises additional challenges for the anonymity of users in digital communication. Nowadays, service providers can access data of all kinds of their users, ranging from metadata to contents of a communication. The exploitation of metadata—in particular about communication relationships—is only just beginning. Service providers like Facebook are in a wealth of personal data, especially metadata. They are expected to utilize these data systematically for monetization and discovery of additional business cases (if not done today); government agencies may also use this information for their purposes like understanding and predicting social movements. Even today, the data usage without the users' consent runs rampant, often already causing undesirable consequences for users [14, 89, 95, 144].

This thesis tackled the challenge of efficient anonymous group communication. With newly arising heterogeneity and ubiquity, *efficiency raises to a considerable challenge* that has to be considered when designing and developing anonymous communication systems. This thesis proposed a novel ACO-based overlay establishment mechanism that connects senders and recipients while considering both efficiency and anonymity as requirements. With efficiency in mind, also *sender anonymity* has to be rethought. This thesis proposed a new mechanism to blend the sender in groups of subjects with configurable size.

To address the challenges arisen from the motivation in Chapter 1, this thesis introduced in Chapter 2 background information that is necessary to follow the contributions of this thesis. The core requirements *efficiency* and *anonymity* were defined; in the case of anonymity, a clear distinction of the often confused terms anonymity, confidentiality, and privacy was provided. The underlying communication model was also presented in Chapter 2 and was complemented with an adversary model that is backed with real-world adversary examples that manifest themselves by combinations of described adversary properties. Then, this background chapter discussed several metrics that aim at measuring and evaluating anonymity. With the communication model, the adversary model, and anonymity metrics at hand, a selection of anonymization primitives was derived and introduced.

In Chapter 3, this thesis provided a selection of characteristics to describe properties of anonymous communication systems concerning efficiency, anonymity, and their general structure. These properties were combined with the previously established anonymization primitives and used to introduce and compare the state of the art in anonymous communication systems.

EFFICIENT AND ANONYMOUS GROUP COMMUNICATION

Chapter 4 proposed a novel ACO-based overlay establishment mechanism. After an introduction to the ACO optimization heuristic, the chapter elaborated on the three-phase approach to establishing communication overlays:

1. *Recipients create and emit agents* that explore the communication system *to find senders*. These agents are being sent through the system and *perform a random walk*. The random walk *decisions to select the next hop are biased* with

pheromones that are deposited by preceding agents in phase 2. Using these pheromones, *agents can consolidate their paths* and *improve the overall efficiency* in the communication system.

2. Agents that found a sender *return on the same path* back to their recipient. On this return, *agents mark the used connections with pheromones* which will bias the random walk of subsequent agents in phase 2.

3. The paths exhibiting the *strongest pheromone trails will be* "activated" and *persisted as the communication overlay*.

In a simulation study, it was shown that the communication overlays that were created using the novel ACO-based communication establishment mechanism require up to 40% fewer connections to provide a connected communication overlay to the senders and recipients. Compared to conventional communication overlays, ACO-based communication overlays do not enforce the usage of shortest path—and increase the communication delays by 9% to 21%—or up to two additional hops between sender and recipients.

While efficiency and anonymity are conflicting optimization goals, the efficiency improvement of ACO-based communication overlays does not impede the achieved anonymity. The anonymity degrees stay stable at 0.996 on social networks, 0.722 on random networks, and 0.773 on RGGs.

INCREASED ROBUSTNESS IN DYNAMIC ENVIRONMENTS Chapter 5 complemented the ACO-based overlay establishment mechanism with approaches to cope with subject churn.

In application scenarios with ubiquitous communication, a stable environment cannot be assumed. Subjects may join and leave the communication system at any point in time, i.e., the communication overlays are exposed to subject *churn* [66, 138, 151, 152].

The ACO-based overlay establishment mechanism facilitates the handling of churn events by distributing knowledge using pheromones—this knowledge can be utilized to repair overlay disruptions that were caused by leaving subjects. Compared to conventional overlays, the established ACO-based communication overlays are more robust when exposed to churn and keep up to 90% of the recipients of recipients connected to the communication overlay while 20% of subjects were leaving due to churn.

routing in the tension of efficiency and anonymity Also in Chapter 5, this thesis reasoned that churn-induced disruptions—as well as the overlay activation in phase 3 of the ACO-based overlay establishment mechanism—require frequent sharing of routing information. Routing information contains knowledge about senders, recipients, and the structure of the communication overlay. With that knowledge, it increases the attack surface for an adversary that aims to identify senders and recipients of a communication. The discussion of behavior under churn is therefore complemented with a discussion of four different mechanisms to establish and share routing information: Successor Lists and Successor Lists with multi-layer encryption—both included in the messages, Bloom filters that carry the routing information, and distributed lookup tables that are stored locally at the subjects. Both successor lists with multi-layer encryption and Bloom filters protect anonymity but rise the communication overhead due to the inclusion of routing information in every message. Distributed lookup tables store the routing information at the subjects, therefore, improve the communication overhead as only a comparably small topic-identifier has to be included in the messages; also, distributed lookup tables protect anonymity by avoiding further sharing of routing information.

Chapter 5 concluded that providing mechanisms to cope with subject churn is facilitated with ACO-based communication overlays and the sharing of routing information can be established with restricted information leakage.

efficient and effective sender protection In Chapter 6, this thesis emphasized the special role of sender anonymity and underlined the associated challenge.

The challenge of sender anonymity is often tackled with approaches that are based upon cover traffic. This chapter first analyzed the influence of *cover traffic randomization* on the efficiency and anonymity and concluded that randomization enables the adversary to remove the cover traffic eventually. With that, the adversary can derive and identify the actual sender.

Therefore, to provide sender anonymity, this chapter proposed a novel approach to provide *sender anonymity* based on ADCnets. Using ADCnets, senders blend into a group of other subjects that use encryption based on multi-party computation: they establish pairwise keys and share their ADCnet-messages with a rendezvous point. Similar to classical DCnets, the sender in the

ADCnet includes their message in the ADCnet-message while the others include nothing ("null") in their message. A so-called rendezvous subject is then able to derive the actual message—without being able to pinpoint the actual sender—and relays the message into the communication overlay; the transmission towards the recipients follows the general communication model.

Chapter 6 reasoned about the anonymity of this scheme by showing that the security proofs of DCnets still hold. Also, efficiency was analyzed, concluding that the communication overhead linearly depends on the size of the ADCnet. The occasional ADCnet establishment is more expensive; the costs for the group formation grow linear with the size of the ADCnets, the costs for the key establishment grow quadratic with the size of the ADCnets.

With the ADCnets-based approach, this thesis proposed in this chapter a scheme for configurable sender anonymity that is efficient and effective.

7.2 OUTLOOK

This thesis proposed novel mechanisms to establish communication overlays and to provide sender anonymity while considering both efficiency and anonymity as compulsory optimization goals. The thesis also goes beyond the state of the art by analyzing and diminishing the impact of subject churn on the robustness of the anonymous communication system that is inherent in dynamic and realistic environments. These contributions advanced the research on anonymous group communication in ubiquitous, heterogeneous, and dynamic environments enabling user anonymity without lower costs for the system and all subjects.

Nonetheless, various aspects and possibilities for further research remain open and will be briefly discussed in the following.

EXISTING NETWORKS MAY NOT BE GOOD ENOUGH Today's systems mostly depend on bootstrapping from existing structures, for example, systems use the friend structures in an Online Social Network (OSN) or real-world relations to initialize the connections in their anonymous communication system. However, it is known that graph-structures can be used to identify communities and even users across different graph-structured datasets [9, 75].

This research in mind, it should be analyzed to which extent information about those structures can be used as evidence to learn about communication structures in an anonymous communication system. In that case, it needs to be studied how logical underlays can be assembled such that they do not leak information; moreover, then it is also inevitable to develop mechanisms that impede the adversaries' efforts to attack already this assembly process.

ESTABLISHING AN APPROPRIATE USER MODEL This thesis used state of the art churn models to predict session durations and intersession times. However, these models were derived from file-sharing networks where the temporal user behavior was observed.

Facing today's vastly different types of devices that may cooperate in such an anonymous communication system, there is a need for new appropriate user models that incorporate the diversity and specifics of devices. Apparent target systems may contain today's anonymous communication systems. Nonetheless, the particular "not-anonymous" equivalent application scenario should also be considered; targeting only anonymous communication systems may introduce a bias towards privacy-aware users which likely are not representative of the whole population.

COPING WITH THE USERS Churn has a manifold influence on a communication system. As shown in this thesis, leaving users may disrupt communication overlays; also, joining users may have to be connected to the communication overlay—or they may even yield substantial improvements for the efficiency and anonymity of the system.

The repair of communication overlays to bypass disruptions may lead to significantly different communication overlays. As such, a global adversary like the service provider will be able to analyze to persistent users and, thereby, reduce the provided anonymity with an intersection attack. As a result of this, the influence of churn also has to be considered from the perspective of an adversary trying to identify senders and recipients.

Similarly, joining users may yield the possibility for more efficient or more anonymous communication overlay. However, for that, the overlay has to be reconsidered. Without global knowledge, this reconsideration will likely lead to a new phase of overlay establishment. As users may continuously join (and leave), the system will remain in a constant overlay establishment stage. As such, it has to be analyzed how—and when—to utilize joining subjects to improve a communication overlay.

UTILIZE ANONYMOUS COMMUNICATION FOR ANONYMOUS DATA PUBLISHING The usage of the anonymous communication system opens the opportunity to request information from subject in the anonymous communication system. The subjects may cooperate and anonymize the supplied data iteratively to answer such requests with statistically correct but anonymous information.

Google [52] and Apple [6] did first steps in this direction. Their idea of anonymous data collection may stimulate research in distributed anonymous data publishing. With that, service providers may realize that they can operate their business without relying on the usage of their users' data.

INCENTIVIZE SERVICE PROVIDERS TO OFFER ANONYMOUS COMMUNICATION Fully decentralized OSNs aimed at removing the service provider as central entity; prominent representatives were Safebook [31, 32] and Diaspora*[1]. Mostly, they failed to succeed; Diaspora*, for example, is said to have about 670,000 users as of now [142]. The number of users stagnates; in comparison to that, Facebook had 2,2 billion users in the last quarter of 2017 and grew steadily over the last years [2]. One of the essential factors for their failure is the so-called network effect [81]; without considerable "start" population or interest, a network offers limited benefit for new users—which in turn deters other users and so on.

With the idea of distributed anonymous data publishing in mind, it is reasonable to think about hybrid services; in these hybrid services, service providers offer their service and also an anonymity-preserving realization based on the cooperation of their users. Their business model that relies on the collected data may then be realized with the help of the anonymously collected information.

[1] https://diasporafoundation.org

[2] Source: https://www.statista.com

For this approach of incentivizing service providers to offer their service also with anonymity, interdisciplinary research is required. Information systems and economics research need to reveal how to transform today's business models to novel ones that rely only on the utilization of anonymous data. Computer science, therefore, has to ensure that the anonymous data provides enough utility to realize those business models.

APPENDIX

A NOMENCLATURE AND ABBREVIATIONS

This chapter summarizes the nomenclature and abbreviations used throughout this thesis. First, the common nomenclature of all chapters is summarized. After that, the specifics of each of the chapters are summarized.

Table A.26: Nomenclature and Abbreviations used throughout this Thesis

Abbreviation	Instances	Meaning
G	-	Logical Underlay
O_t	-	Communication overlay for topic/interest t
$\mathcal{V}$	$\{v_1, \dots, v_n\}$	Vertices, the subjects in the system
$\mathcal{V}_t \subseteq \mathcal{V}$	-	Subjects, participating in O_t
$\mathcal{E}$	$\{e_1, \dots, e_n\}$	Edges, the connections in the system
$\mathcal{E}_t \subseteq \mathcal{E}$	-	Edges, used to connect subjects in O_t
T	$\{t_1, \dots, t_n\}$	Topics, the interests of comm. groups
RD	$\{rd_1, \dots, rd_n\}$	Rounds, the time steps that order the events
$d^{+/-}(v)$	-	Incoming (d^+)/outgoing (d^-) degree of v
$N^{+/-}(v)$	-	Incoming (N^+)/outgoing (N^-) neighbors of v
$p(v_k, v_l)$	$(e_1, \dots, e_n)$	Path between v_k and v_l
$sp(v_k, v_l)$	-	Shortest possible path between v_k and v_l
$\lvert p(v_k, v_l)\rvert$	-	Length of a path between v_k and v_l, the number of connections in the sequence
$apl(G)$	-	Average shortest path length $\lvert sp(v_k, v_l)\rvert$ between any connected pair of subjects
$diam(G)$	-	Diameter of G, the longest shortest path between any pair of subjects
$w_{\mathcal{E}}(G)$	-	Weight of G, the number of connections
k_{bf}	-	Number of hash functions of the bloom filter
m_{bf}	-	Size of the bloom filter
$\mathcal{S}_t$	$\{s_1, \dots, s_n\}$	Sources, the *senders* of information
$\mathcal{R}_t$	$\{r_1, \dots, r_n\}$	Receivers, the *recipients* of information
$\mathcal{B}_t$	$\{b_1, \dots, b_n\}$	*Brokers*, the forwarders of information
m_t^{com}	-	Communication messages for topic t
$\mathfrak{A}$	$\mathfrak{A}_{p,-}$, $\mathfrak{A}_{a,-}$	*p*assive or *a*ctive Adversary
	$\mathfrak{A}_{-,l}$, $\mathfrak{A}_{-,c}$, $\mathfrak{A}_{-,g}$	*l*ocal, *c*olluding, or *g*lobal Adversary
$\mathfrak{C}$	$\{c_1, \dots, c_n\}$	Subjects controlled by the adversary

Table A.27: Nomenclature and Abbreviations specific to Efficient and Anonymous Group Communication (Chapter 4)

Abbreviation	Instances	Meaning
$\mathcal{A} = \mathcal{A}_{\mathrm{Disc}} \cup \mathcal{A}_{\mathrm{Ret}}$	$\{a_1, \ldots, a_n\}$	Agents, establishing the communication overlay
$\mathcal{A}_{\mathrm{Disc}}$	-	Agents, looking for senders
$\mathcal{A}_{\mathrm{Ret}}$	-	Agents, returning to recipients
τ_{v_i}	-	Pheromones, placed at the connection towards subject v_i
τ_{e_i}	-	Pheromones, placed at connection e_i
$a_i.\sigma$	-	Agent a_i's strictness

Table A.28: Nomenclature and Abbreviations specific to Routing in the Tension of Efficiency and Anonymity (Chapter 5)

Abbreviation	Instances	Meaning
sk	sk_{v_l,v_k}	Symmetric key shared between v_l and v_k
$enc_{\mathrm{blocksize}}$		Block size of the cryptographic algorithm

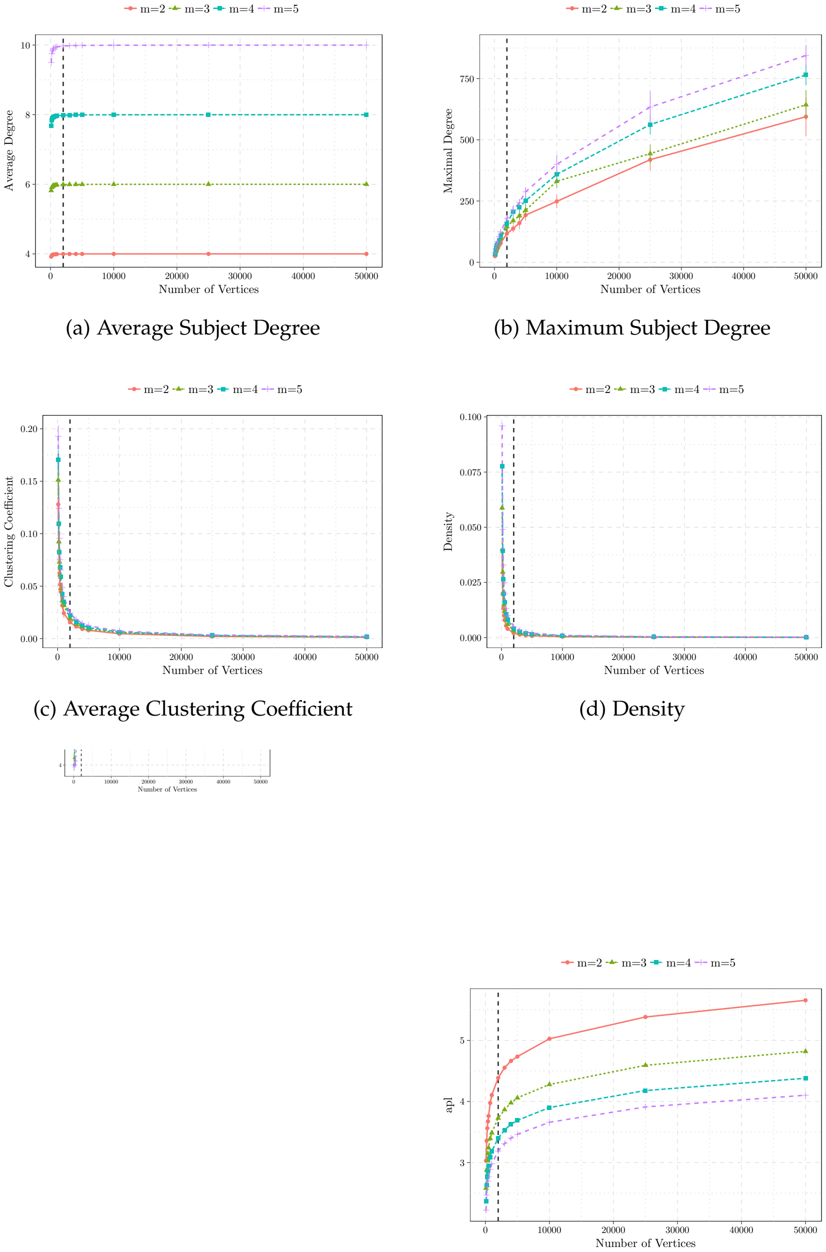
Table A.29: Nomenclature and Abbreviations specific to Efficient and Effective Sender Protection (Chapter 6)

Abbreviation	Instances	Meaning
m_t^{cov}		Cover message; Noise
k_{ADCnet}		Size of ADCnets
sk	sk_{v_l,v_k}^{rd}	session key shared between v_l and v_k for round $rd \in RD$

B

NETWORK CONFIGURATION

B.1 SOCIAL NETWORK



(a) Average Subject Degree

(b) Maximum Subject Degree

(c) Average Clustering Coefficient

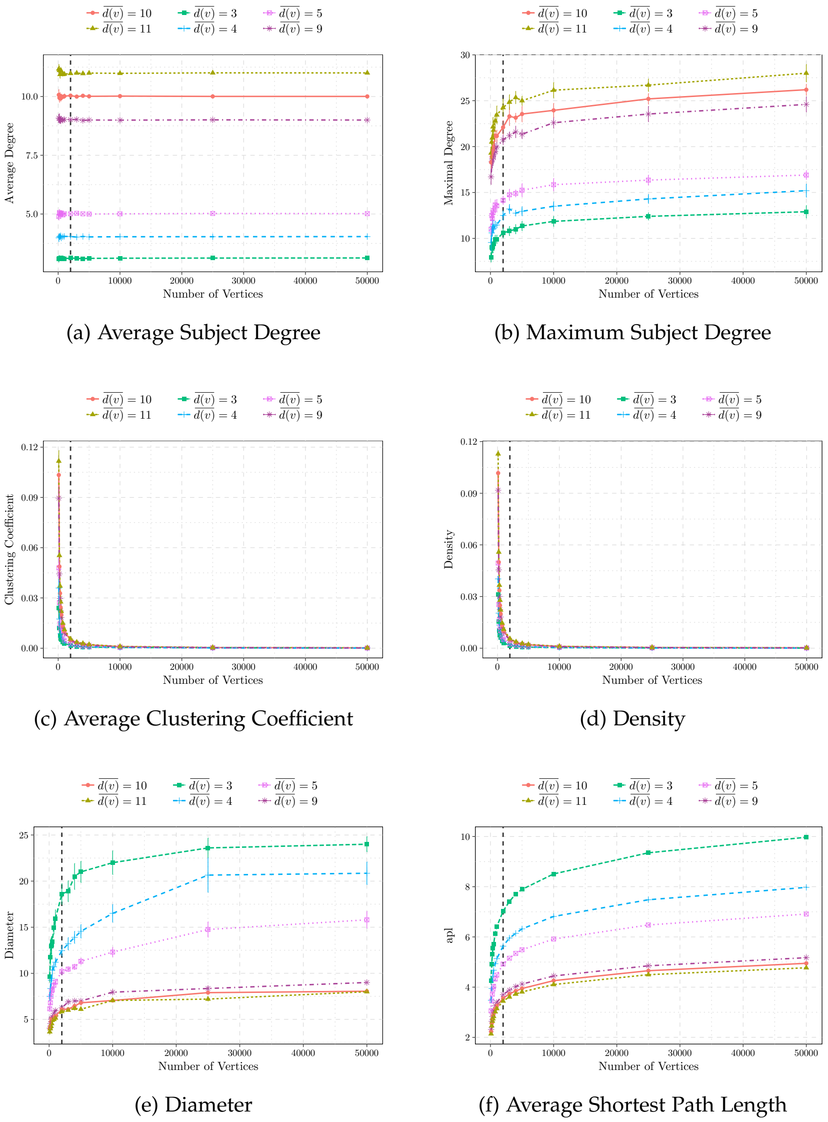
(d) Density

(e) Diameter

(f) Average Shortest Path Length

Figure B.57: Graph-theoretic properties of social networks following the Barabàsi-Albert model. The gray, dashed vertical line marks the selected network size of 2,000 subjects.

B.2 RANDOM NETWORK



(a) Average Subject Degree

(b) Maximum Subject Degree

(c) Average Clustering Coefficient

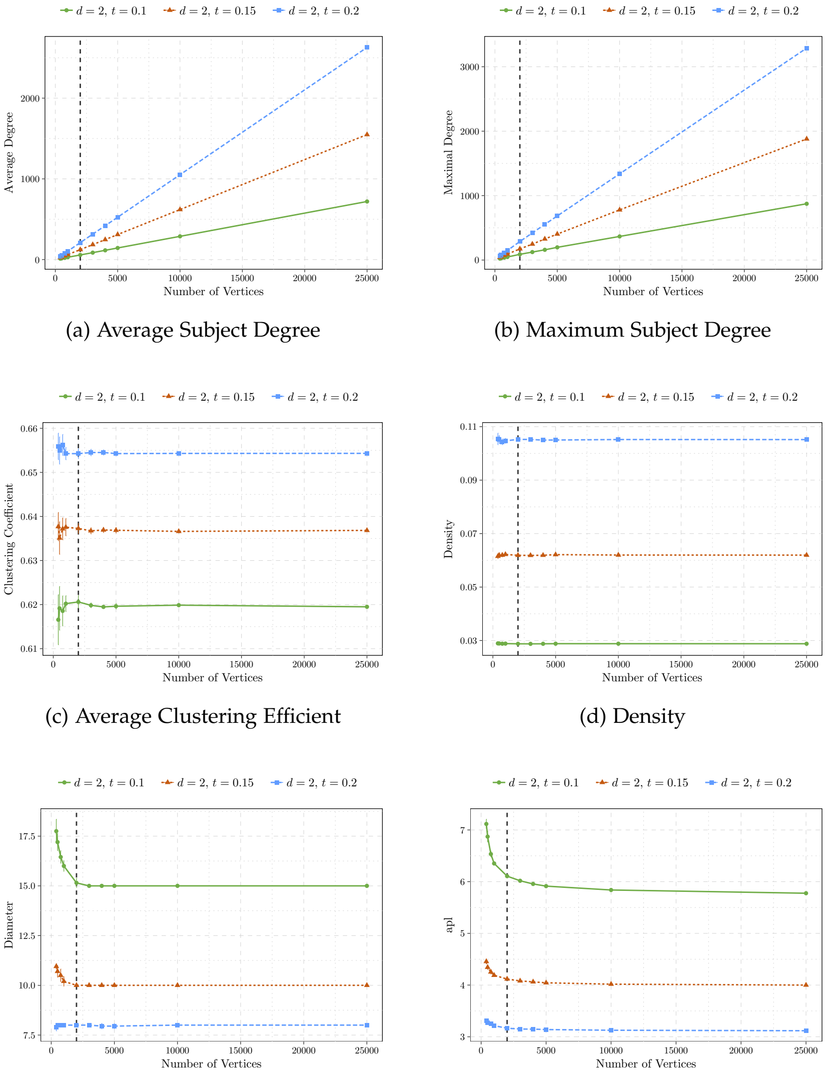
(d) Density

(e) Diameter

(f) Average Shortest Path Length

Figure B.58: Graph-theoretic properties of random networks. The gray, dashed vertical line marks the selected network size of 2,000 subjects.

B.3 RANDOM GEOMETRIC GRAPH (RGG)



(a) Average Subject Degree

(b) Maximum Subject Degree

(c) Average Clustering Efficient

(d) Density

(e) Diameter

(f) Average Shortest Path Length

Figure B.59: Graph-theoretic properties of RGGs. The gray, dashed vertical line marks the selected network size of 2,000 subjects.

C

BLOOM FILTER CONFIGURATIONS

The probability of false positive hits of a Bloom filter follows from

$$p_{fp} = (1 - e^{mk/n})^{k},$$

if the Bloom filter is configured with m being the size of the Bloom filter, n the number of entries that are added to the Bloom filter, and k being the number of hash functions that are used to fill the Bloom filter. The ratio m/n provides then the number of bits per entry in the Bloom filter. Table C.30 lists pre-calculated false positive probabilities for $m/n \in [0, 20]$ and $k \in [0, 20]$

Table C.30: False positive probabilities in dependency of size m, number of entries added n, and number of hash functions k.

		k–number of hash functions used						
		1	2	3	4	5	6	7
m/n–number of bits per entry added	1	0.6321	0.7476	0.8580	0.9287	0.9668	0.9852	0.9936
	2	0.3935	0.3996	0.4689	0.5590	0.6516	0.7361	0.8068
	3	0.2835	0.2368	0.2526	0.2941	0.3511	0.4179	0.4897
	4	0.2212	0.1548	0.1469	0.1597	0.1849	0.2198	0.2628
	5	0.1813	0.1087	0.0918	0.0920	0.1009	0.1164	0.1378
	6	0.1535	0.0804	0.0609	0.0561	0.0578	0.0638	0.0734
	7	0.1331	0.0618	0.0423	0.0359	0.0347	0.0364	0.0403
	8	0.1175	0.0489	0.0306	0.0240	0.0217	0.0216	0.0229
	9	0.1052	0.0397	0.0228	0.0166	0.0141	0.0133	0.0135
	10	0.0952	0.0329	0.0174	0.0118	0.0094	0.0084	0.0082
	11	0.0869	0.0276	0.0136	0.0086	0.0065	0.0055	0.0051
	12	0.0800	0.0236	0.0108	0.0065	0.0046	0.0037	0.0033
	13	0.0740	0.0203	0.0088	0.0049	0.0033	0.0026	0.0022
	14	0.0689	0.0177	0.0072	0.0038	0.0024	0.0018	0.0015
	15	0.0645	0.0156	0.0060	0.0030	0.0018	0.0013	0.0010
	16	0.0606	0.0138	0.0050	0.0024	0.0014	0.0009	0.0007
	17	0.0571	0.0123	0.0042	0.0019	0.0011	0.0007	0.0005
	18	0.0540	0.0111	0.0036	0.0016	0.0008	0.0005	0.0004
	19	0.0513	0.0100	0.0031	0.0013	0.0007	0.0004	0.0003
	20	0.0488	0.0091	0.0027	0.0011	0.0005	0.0003	0.0002

		k–number of hash functions used						
		8	9	10	11	12	13	14
m/n–number of bits per entry added	1	0.9973	0.9989	0.9995	0.9998	0.9999	1.0000	1.0000
	2	0.8625	0.9043	0.9346	0.9560	0.9707	0.9806	0.9873
	3	0.5621	0.6315	0.6954	0.7521	0.8011	0.8422	0.8761
	4	0.3125	0.3670	0.4246	0.4835	0.5418	0.5980	0.6510
	5	0.1646	0.1967	0.2336	0.2748	0.3194	0.3667	0.4155
	6	0.0865	0.1031	0.1233	0.1471	0.1747	0.2056	0.2398
	7	0.0463	0.0544	0.0646	0.0772	0.0923	0.1101	0.1306
	8	0.0255	0.0292	0.0342	0.0405	0.0483	0.0578	0.0691
	9	0.0145	0.0161	0.0184	0.0215	0.0254	0.0303	0.0362
	10	0.0085	0.0091	0.0102	0.0116	0.0136	0.0160	0.0190
	11	0.0051	0.0053	0.0058	0.0064	0.0074	0.0085	0.0100
	12	0.0031	0.0032	0.0033	0.0036	0.0041	0.0046	0.0054
	13	0.0020	0.0019	0.0020	0.0021	0.0023	0.0026	0.0029
	14	0.0013	0.0012	0.0012	0.0012	0.0013	0.0015	0.0016
	15	0.0009	0.0008	0.0007	0.0007	0.0008	0.0008	0.0009
	16	0.0006	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005
	17	0.0004	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003
	18	0.0003	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002
	19	0.0002	0.0002	0.0001	0.0001	0.0001	0.0001	0.0001
	20	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001

		k–number of hash functions used					
		15	16	17	18	19	20
m/n–number of bits per entry added	1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	2	0.9917	0.9946	0.9965	0.9978	0.9986	0.9991
	3	0.9036	0.9255	0.9428	0.9563	0.9668	0.9749
	4	0.6998	0.7440	0.7833	0.8178	0.8478	0.8735
	5	0.4649	0.5138	0.5616	0.6073	0.6506	0.6909
	6	0.2767	0.3159	0.3568	0.3988	0.4413	0.4836
	7	0.1538	0.1798	0.2083	0.2390	0.2719	0.3064
	8	0.0823	0.0976	0.1151	0.1347	0.1565	0.1803
	9	0.0433	0.0517	0.0616	0.0730	0.0861	0.1009
	10	0.0227	0.0271	0.0324	0.0387	0.0460	0.0546
	11	0.0119	0.0142	0.0170	0.0203	0.0243	0.0289
	12	0.0063	0.0075	0.0089	0.0106	0.0127	0.0152
	13	0.0034	0.0040	0.0047	0.0056	0.0067	0.0080
	14	0.0019	0.0021	0.0025	0.0030	0.0035	0.0042
	15	0.0010	0.0012	0.0014	0.0016	0.0019	0.0022
	16	0.0006	0.0006	0.0007	0.0009	0.0010	0.0012
	17	0.0003	0.0004	0.0004	0.0005	0.0005	0.0006
	18	0.0002	0.0002	0.0002	0.0003	0.0003	0.0003
	19	0.0001	0.0001	0.0001	0.0001	0.0002	0.0002
	20	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001

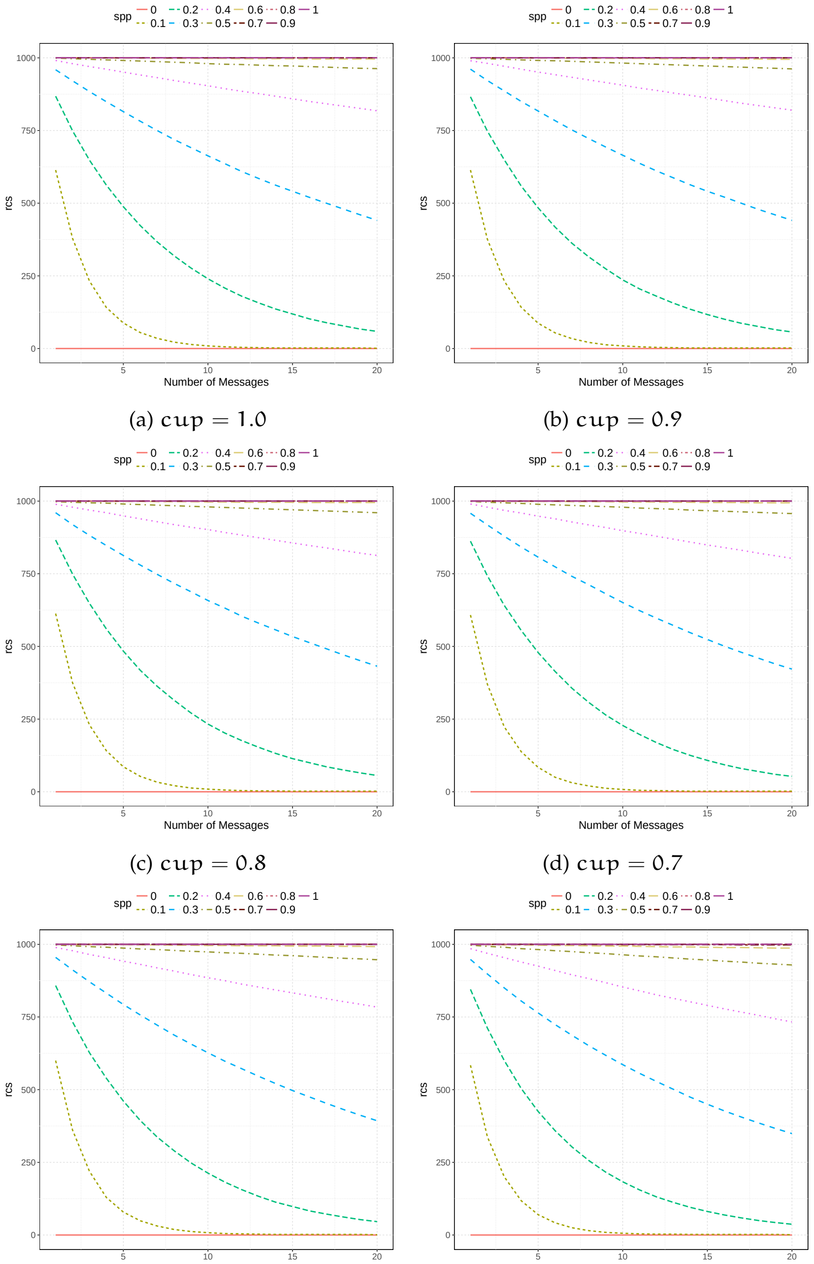
D COVER TRAFFIC: ANONYMITY OVER TIME

Randomized subject participation and connection utilization enable an adversary $\mathfrak{A}$ to reduce the set of cover subjects over time.

This chapter presents extended results of the conducted simulation study of Section 6.4 that substantiate the conclusion of the inability to improve the efficiency of cover traffic by randomized initialization.

Figure D.60 shows the results of the simulations in which the connection utilization probability cup is fixed with $\{1.0, 0.9, 0.8, 0.7, 0.6, 0.5\}$; Figure D.61 shows the respective plots in which the connection utilization probability cup is fixed with $\{0.4, 0.3, 0.2, 0.1\}$.

If considerable efficiency improvement is achieved with configuring the subject participation probability spp and the connection utilization probability cup at 0.5 or lower, the anonymity over degrades with every communication message that is sent.



(a) $\mathrm{cup} = 1.0$

(b) $\mathrm{cup} = 0.9$

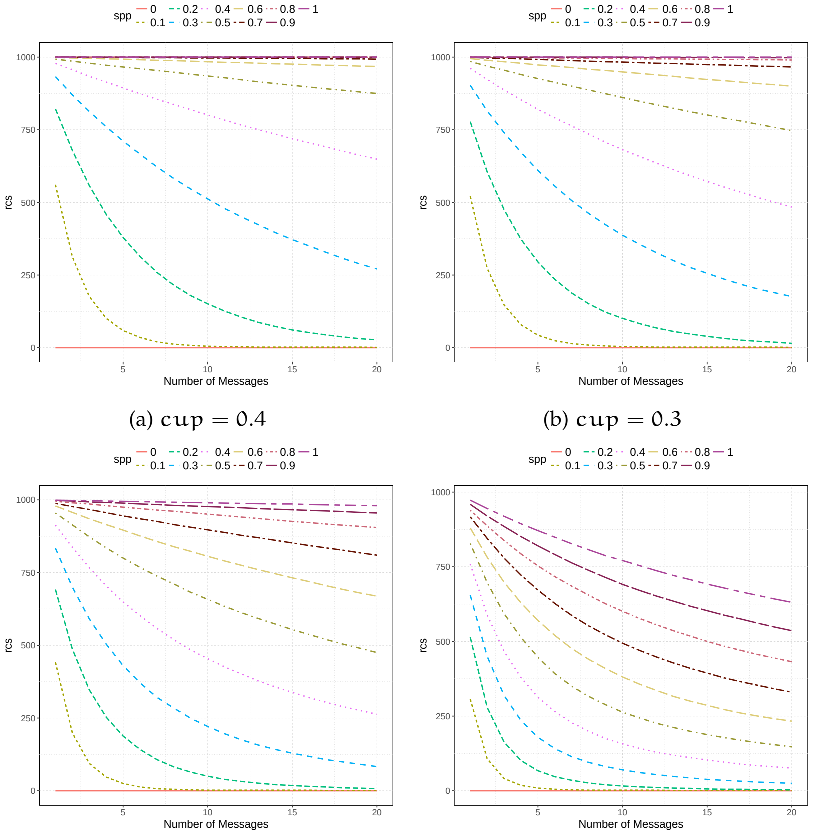
(c) $\mathrm{cup} = 0.8$

(d) $\mathrm{cup} = 0.7$

(e) $\mathrm{cup} = 0.6$

(f) $\mathrm{cup} = 0.5$

Figure D.60: Number of Cover Nodes over multiple Messages with varying spp probabilities. The cup probability is fixed between 1.0 and 0.5 in each plot.



(a) cup = 0.4

(b) cup = 0.3

(c) cup = 0.2

(d) cup = 0.1

Figure D.61: Number of Cover Nodes over multiple Messages with varying spp probabilities. The cup probability is fixed between 0.4 and 0.1 in each plot.

BIBLIOGRAPHY

1. Albert, R. & Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of Modern Physics* **74,** 47–97. ISSN: 1478-3967 (2002).
2. AlSabah, M., Bauer, K., Elahi, T. & Goldberg, I. *The path less travelled: Overcoming Tor's bottlenecks with traffic splitting* in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **7981 LNCS** (2013), 143–163. ISBN: 9783642390760. doi:10.1007/978-3-642-39077-7_8.
3. AlSabah, M., Bauer, K. & Goldberg, I. Enhancing Tor's performance using real-time traffic classification. *CCS '12 Proceedings of the 2012 ACM conference on Computer and communications security,* 73–84. ISSN: 15437221 (2012).
4. Anonymizer Inc. *The Anonymizer* 1995. <https://www.anonymizer.com>.
5. Apel, S. & Buchmann, E. Biology-Inspired Optimizations of Peer-to-Peer Overlay Networks. *Praxis der Informationsverarbeitung und Kommunikation* **28,** 199–205. ISSN: 0930-5157 (2005).
6. Apple. *Learning with Privacy at Scale* 2017. <https://machinelearning.apple.com/2017/12/06/learning-with-privacy-at-scale.html> (visited on 03/08/2017).
7. Backes, M., Kate, A., Meiser, S. & Mohammadi, E. *(Nothing else) MATor(s): Monitoring the Anonymity of Tor's Path Selection* in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)* (ACM, 2014), 513–524. ISBN: 9781450329576. doi:10.1145/2660267.2660371.
8. Backes, M., Meiser, S. & Slowik, M. Your Choice MATor(s). *Proceedings on Privacy Enhancing Technologies* **2016.** ISSN: 2299-0984. doi:10.1515/popets-2016-0004. <https://www.degruyter.com/view/j/popets.2015.2016.issue-2/popets-2016-0004/popets-2016-0004.xml> (2016).

9. Backstrom, L., Dwork, C. & Kleinberg, J. *Wherefore art thou r3579x?* in *Proceedings of the 16th international conference on World Wide Web - WWW '07* (2007), 181. ISBN: 9781595936547. doi:10.1145/1242572.1242598. <http://portal.acm.org/citation.cfm?doid=1242572.1242598>.

10. Bacon, J., Eyers, D. M., Singh, J. & Pietzuch, P. R. Access control in publish/subscribe systems. *Proceedings of the second international conference on Distributed event-based systems - DEBS '08,* 23 (2008).

11. Bagci, H., Korpeoglu, I. & Yazici, A. A Distributed Fault-Tolerant Topology Control Algorithm for Heterogeneous Wireless Sensor Networks. *IEEE Parallel and Distributed Systems* **26,** 914–923. ISSN: 1045-9219 (2015).

12. Barabási, A.-L. *Network Science* 475. ISBN: 9781107076266. <http://barabasi.com/book/network-science> (2016).

13. Barabási, A.-L. & Albert, R. Emergence of Scaling in Random Networks. *Science* **286,** 509–512. ISSN: 00368075 (1999).

14. BBC. *Ftiness app Strava lights up staff at military bases* Jan. 2018. <http://www.bbc.com/news/technology-42853072> (visited on 02/12/2012).

15. Berthold, O., Federrath, H. & Köpsell, S. in *Designing Privacy Enhancing Technologies* (Springer, 2001). ISBN: 978-3-540-41724-8, 978-3-540-44702-3. doi:10.1007/3-540-44702-4_7.

16. Bloom, B. H. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM* **13,** 422–426. ISSN: 00010782 (July 1970).

17. Blum, C. Ant colony optimization : Introduction and recent trends. *Physics of Life Reviews* **2,** 353–373. ISSN: 1571-0645 (2005).

18. Böck, L., Karuppayah, S., Grube, T., Mühlhäuser, M. & Fischer, M. *Hide and seek: Detecting sensors in P2P botnets* in *2015 IEEE Conference on Communications and Network-Security, CNS 2015* (2015), 731–732. ISBN: 9781467378765. doi:10.1109/CNS.2015.7346908.

19. Borges, F., Buchmann, J. & Muhlhauser, M. *Introducing asymmetric DC-nets* in *2014 IEEE Conference on Communications and Network Security, CNS 2014* (2014), 508–509. ISBN: 9781479958900. doi:10.1109/CNS.2014.6997528.

20. Boyko, V., MacKenzie, P. & Patel, S. *Provably Secure Password-Authenticated Key Exchange Using Diffie-Hellman* in *Advances in Cryptology - EUROCRYPT 2000* (Springer-Verlag Berlin Heidelberg, 2000), 156–171. doi:https://doi.org/10.1007/3-540-45539-6_12.

21. Bundesgesetzblatt. *Gesetz zur Beschränkung des Brief-, Post- und Fernmeldegeheimnisses (Artikel 10-Gesetz - G 10) i.d.F. von 17. August 2017* 2001. <http://www.bgbl.de/xaver/bgbl/start.xav?startbk=Bundesanzeiger%7B%5C_%7DBGBl%7B%5C&%7DjumpTo=bgbl101s1254.pdf>.

22. Chaum, D. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 65–75. ISSN: 09332790 (1988).

23. Chaum, D. L. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM* **24,** 84–90. ISSN: 00010782 (1981).

24. Chaum, D., Javani, F., Krasnova, A., Kate, A., de Ruiter, J. & Sherman, A. T. *cMix: Mixing with Minimal Real-Time Asymmetric Cryptographic Operations* in *International Conference on Applied Cryptography and Network Security (ACNS)* (Springer, 2017). ISBN: 1234567245. doi:10:47510.1007/978-3-319-61204-1_28/123_4.

25. Cohen, R. & Havlin, S. Scale-Free Networks are Ultrasmall. ISSN: 0031-9007. doi:10.1103/PhysRevLett.90.058701. arXiv: 0205476 [cond-mat]. <http://arxiv.org/abs/cond-mat/0205476%7B%%7D0Ahttp://dx.doi.org/10.1103/PhysRevLett.90.058701> (2002).

26. Congress of the United States of America. *Foreign Intelligence Surveillance Act of 1978 (Public Law 95-511)* 1978. <https://www.gpo.gov/fdsys/pkg/STATUTE-92/pdf/STATUTE-92-Pg1783.pdf>.

27. Congress of the United States of America. *Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism (USA PATRIOT ACT) Act of 2001* 2011. <https://www.gpo.gov/fdsys/pkg/STATUTE-115/pdf/STATUTE-115-Pg272.pdf>.

28. Congress of the United States of America. *Uniting and Strengthening America by Fulfilling Rights and ensuring Effective Discipline Over Monitoring Act of 2015 (USA FREE-*

DOM Act) 2015. <https://www.congress.gov/114/bills/hr2048/BILLS-114hr2048enr.pdf>.

29. Corrigan-Gibbs, H., Boneh, D. & Mazieres, D. *Riposte: An anonymous messaging system handling millions of users* in *IEEE Symposium on Security and Privacy (S&P)* (2015), 321–338. ISBN: 9781467369497. doi:10.1109/SP.2015.27.

30. Corrigan-Gibbs, H. & Ford, B. *Dissent: Accountable Anonymous Group Messaging* in *Proceedings of the 17th ACM conference on Computer and communications security (CCS)* (ACM, 2010). ISBN: 9781450302456. doi:10.1145/1866307.1866346.

31. Cutillo, L. A., Molva, R. & Önen, M. *Safebook: A distributed privacy preserving online social network* in *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2011 - Digital Proceedings* (2011). ISBN: 9781457703515. doi:10.1109/WoWMoM.2011.5986118.

32. Cutillo, L. A., Molva, R. & Strufe, T. Safebook: A privacy-preserving online social network leveraging on real-life trust. *IEEE Communications Magazine.* ISSN: 01636804. doi:10.1109/MCOM.2009.5350374 (2009).

33. Damgard, I. B. *Collision free hash functions and public key signature schemes* in *Lecture Notes in Computer Science* (eds Chaum, D. & Price, W. L.) **304 LNCS** (1988), 203–216. ISBN: 9783540191025. doi:10.1007/3-540-39118-5_19.

34. Danezis, G., Dingledine, R. & Mathewson, N. *Mixminion: Design of a type III anonymous remailer protocol* in *Proceedings - IEEE Symposium on Security and Privacy* (IEEE, 2003), 2–15. ISBN: 0769519407. doi:10.1109/SECPRI.2003.1199323.

35. Danezis, G. *Measuring anonymity: a few thoughts and a differentially private bound* in *Proceedings of the DIMACS Workshop on Measuring Anonymity* (2013), 1–10.

36. Danezis, G. & Serjantov, A. *Statistical Disclosure or Intersection Attacks on Anonymity Systems* in *Information Hiding* **3200** (Springer, 2005), 293–308. ISBN: 978-3-540-24207-9. doi:10.1007/978-3-540-30114-1_21.

37. Daubert, J., Fischer, M., Grube, T., Schiffner, S., Kikiras, P. & Mühlhäuser, M. AnonPubSub: Anonymous publish-subscribe overlays. *Computer Communications* **76,** 42–53. ISSN: 01403664 (2016).

38. Daubert, J., Fischer, M., Schiffner, S. & Mühlhäuser, M. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 685–691 (Springer, 2013). ISBN: 9783642386305. doi:10.1007/978-3-642-38631-2_57.

39. Daubert, J., Grube, T., Mühlhäuser, M. & Fischer, M. *Internal attacks in anonymous publish-subscribe P2P overlays* in *Proceedings - International Conference on Networked Systems, NetSys 2015* (2015). ISBN: 9781479958047. doi:10.1109/NetSys.2015.7089074.

40. Daubert, J., Grube, T., Mühlhäuser, M. & Fischer, M. *On the anonymity of privacy-preserving many-to-many communication in the presence of node churn and attacks* in *2016 13th IEEE Annual Consumer Communications and Networking Conference, CCNC 2016* (2016), 738–744. ISBN: 9781467392921. doi:10.1109/CCNC.2016.7444871.

41. De Montjoye, Y.-A., Radaelli, L., Singh, V. K. & Pentland, A. S. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science* **347,** 536–539. ISSN: 0036-8075 (2015).

42. Di Caro, G. *Ant colony optimization and its application to adaptive routing in telecommunication networks* PhD thesis (Brussels, Belgium, Nov. 2004). <http://theses.ulb.ac.be/ETD-db/collection/available/ULBetd-09172004-201049/>.

43. Di Crescenzo, G., Coan, B., Schultz, J., Tsang, S. & Wright, R. N. in, 114–132 (Springer, Berlin, Heidelberg, 2014). doi:10.1007/978-3-642-54568-9_8. <http://link.springer.com/10.1007/978-3-642-54568-9%7B%5C_%7D8>.

44. Diaz, C., Seys, S., Claessens, J. & Preneel, B. Towards measuring anonymity. *Privacy Enhancing Technologies,* 1–15. ISSN: 16113349 (2003).

45. Diffie, W. & Hellman, M. E. New Directions in Cryptography. *IEEE Transactions on Information Theory* **22,** 644–654. ISSN: 15579654 (1976).

46. Dingledine, R., Mathewson, N. & Syverson, P. *Tor: The second-generation onion router* in *Proceedings of the 13th conference on USENIX Security Symposium (SSYM)* **13** (USENIX

Association, 2004). doi:10.1.1.4.6896. <http://portal.acm.org/citation.cfm?id=1251375.1251396>.

47. Dorigo, M., Caro, G. D. & Gambardella, L. M. Ant Algorithms for Discrete Optimization. *Artificial Life* **5,** 137–172. ISSN: 1064-5462 (Apr. 1999).

48. Edman, M. & Yener, B. On anonymity in an electronic society. *ACM Computing Surveys* **42,** 1–35. ISSN: 03600300 (2009).

49. Eiben, A. E. & Smith, J. E. *Introduction to Evolutionary Computing* 2nd ed. ISBN: 978-3-662-44873-1. doi:10.1007/978-3-662-44874-8 (Springer-Verlag Berlin Heidelberg, 2016).

50. ENISA. *Algorithms, key size and parameters report 2014* tech. rep. (ENISA, 2014).

51. Erdös, P. & Rényi, A. On random graphs. *Publicationes Mathematicae* **6,** 290–297 (1959).

52. Erlingsson, Ú., Pihur, V. & Korolova, A. *RAPPOR* in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14* (2014), 1054–1067. ISBN: 9781450329576. doi:10.1145/2660267.2660348. arXiv: 1407.6981. <http://dl.acm.org/citation.cfm?doid=2660267.2660348>.

53. Eugster, P. T., Felber, P., Guerraoui, R. & Kermarrec, A.-M. The Many Faces of Publish/Subscribe. *ACM Computing Surveys* **35,** 114–131 (2003).

54. European Parliament. *Directive 95/46/EC on the protection of Individuals with regard to the processing of personal data and the free movement of such data)* 1995. <http://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX%7B%%7D3A31995L0046>.

55. European Parliament. *Regulation (EU) 2016/679 (General Data Protection Regulation)* 2016. <http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679>.

56. Evans, N. S., Dingledine, R. & Grothoff, C. A Practical Congestion Attack on Tor Using Long Paths. *18th USENIX Security Symposium* **19,** 33–50 (2009).

57. Fan, L., Cao, P., Almeida, J. & Broder, A. Z. Summary cache: A scalable wide-area Web cache sharing protocol. *IEEE/ACM Transactions on Networking* **8,** 281–293. ISSN: 10636692 (June 2000).

58. Freedman, M. J. & Morris, R. *Tarzan: a peer-to-peer anonymizing network layer* in *Proceedings of the 9th ACM conference on Computer and communications security (CCS)* (2002), 193. ISBN: 1581136129. doi:10.1145/586110.586137.

59. Ganesh, A., Kermarrec, A.-M. & Massoulie, L. Peer-to-peer membership management for gossip-based protocols. *IEEE Transactions on Computers* **52,** 139–149. ISSN: 0018-9340 (Feb. 2003).

60. Gilbert, E. N. Random Graphs. *Annals of Mathematical Statistics* **30,** 1141–1144. ISSN: 0003-4851 (1959).

61. Goel, S., Robson, M., Polte, M. & Sirer, E. *Herbivore: A scalable and efficient protocol for anonymous communication* tech. rep. (Cornell University, 2003), 17.

62. Goldschlag, D., Reed, M. & Syverson, P. in *Information Hiding* 137–150 (Springer, 1996). ISBN: 978-3-540-61996-3. doi:10.1007/3-540-61996-8_37.

63. Golle, P. & Juels, A. Dining Cryptographers Revisited. *Advances in Cryptology - EUROCRYPT 2004,* 456–473 (2004).

64. Greenberg, A. The Father of Online Anonymity Has a Plan to End the Crypto War. *Wired.* <https://www.wired.com/2016/01/david-chaum-father-of-online-anonymity-plan-to-end-the-crypto-wars/> (2016).

65. Greenwald, G. *XKeyscore: NSA tool collects 'nearly everything a user does on the internet'* 2013. <https://www.theguardian.com/world/2013/jul/31/nsa-top-secret-program-online-data> (visited on 03/27/2017).

66. Groß, C., Richerzhagen, B. & Lehn, M. in *Benchmarking Peer-to-Peer Systems* (eds Effelsberg, W., Steinmetz, R. & Strufe, T.) 41–67 (Springer-Verlag Berlin Heidelberg, 2013). ISBN: 978-3-642-38672-5. doi:10.1007/978-3-642-38673-2.

67. Grube, T., Hauke, S., Daubert, J. & Mühlhäuser, M. *Ant colonies for efficient and anonymous group communication systems* in *International Conference on Networked Systems (NetSys)* (IEEE, 2017). ISBN: 9781509043941. doi:10.1109/NetSys.2017.7903958.

68. Grube, T., Hauke, S., Daubert, J. & Mühlhäuser, M. *Ant colony optimisation - A solution to efficient anonymous group communication?* in *2017 14th IEEE Annual Consumer Communications and Networking Conference, CCNC 2017* (2017). ISBN: 9781509061969. doi:10.1109/CCNC.2017.7983129.

69. Grube, T., Schiller, B. & Strufe, T. *Monotone sampling of networks* in *CEUR Workshop Proceedings* **1229** (2014), 37–48.

70. Grube, T., Thummerer, M., Daubert, J. & Mühlhäuser, M. *Cover Traffic: A Trade of Anonymity and Efficiency* in *Security and Trust Management* (Springer Nature, 2017).

71. Grube, T., Volk, F., Mühlhäuser, M., Bhairav, S., Sachidananda, V. & Elovici, Y. *Complexity Reduction in Graphs: A user Centric Approach to Graph Exploration* in *International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC)* (IARIA, 2017).

72. Grube, T., Volk, F., Mühlhäuser, M., Bhairav, S., Sachidananda, V. & Elovici, Y. Complexity Reduction in Graphs: A user Centric Approach to Graph Exploration. *International Journal on Advances in Intelligent Systems* **11** (2018).

73. Gutjahr, W. J. A Graph-based Ant System and its convergence. *Future Generation Computer Systems* **16,** 873–888. ISSN: 0167739X (June 2000).

74. Gutjahr, W. J. ACO algorithms with guaranteed convergence to the optimal solution. *Information Processing Letters* **82,** 145–153. ISSN: 00200190 (2002).

75. Hay, M., Miklau, G., Jensen, D. & Towsley, D. *Resisting structural identification in anonymized social networks* in *Proceedings of the 34th International Conference on very large databases* (2008), 102–114. ISBN: 0000000000000. doi:10.1007/s00778-010-0210-x..

76. Hook, L. *Uber loses ground in US as rival Lyft accelerates* June 2017. <https://www.ft.com/content/b4fb76a6-52dd-11e7-bfb8-997009366969> (visited on 02/12/2018).

77. Hughes, D. & Shmatikov, V. Information Hiding, Anonymity and Privacy: A Modular Approach. *Journal of Computer Security* **12,** 3–36. ISSN: 0926-227X (2004).

78. Johnson, A., Jansen, R., Hopper, N., Segal, A. & Syverson, P. PeerFlow: Secure Load Balancing in Tor. *Proceedings on Privacy Enhancing Technologies* **2017.** ISSN: 2299-0984. doi:10.1515/popets-2017-0017. <http://www.degruyter.com/view/j/popets.2017.2017.issue-2/popets-2017-0017/popets-2017-0017.xml> (2017).

79. Karp, R. M. in *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art* 219–241 (Springer US, Boston, MA, 2010). ISBN: 9783540682745. doi:10.1007/978-3-540-68279-0_8. arXiv: arXiv:1011.1669v3. <http://link.springer.com/10.1007/978-1-4684-2001-2%7B%5C_%7D9>.

80. Karuppayah, S., Manickam, S., Böck, L., Grube, T., Mühlhäuser, M. & Fischer, M. *SensorBuster: On Identifying Sensor Nodes in P2P Botnets* in *International Conference on Availability, Reliability and Security (ARES)* **Part F1305** (IEEE Computer Society, 2017). ISBN: 9781450352574. doi:10.1145/3098954.3098991.

81. Katz, M. L. & Shapiro, C. Systems Competition and Network Effects. *Journal of Economic Perspectives.* ISSN: 0895-3309. doi:10.1257/jep.8.2.93 (1994).

82. Kennedy, J. & Eberhart, R. *Particle swarm optimization* in *Proceedings of ICNN'95 - International Conference on Neural Networks* **4** (IEEE, 1995), 1942–1948. ISBN: 0-7803-2768-3. doi:10.1109/ICNN.1995.488968. <http://ieeexplore.ieee.org/document/488968/>.

83. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by Simulated Annealing. *Science* **220,** 671–680. ISSN: 0036-8075 (1983).

84. Kou, L., Markowsky, G. & Berman, L. A fast algorithm for Steiner trees. *Acta Informatica* **15,** 141–145. ISSN: 00015903 (1981).

85. Krawczyk, H. *Cryptographic extraction and key derivation: The HKDF scheme* in *Lecture Notes in Computer Science* **6223 LNCS** (2010), 631–648. ISBN: 3642146228. doi:10.1007/978-3-642-14623-7_34.

86. Kuhn, F., Wattenhofer, R., Zhang, Y. & Zollinger, A. Geometric Ad-Hoc Routing: Of Theory and Practice. *In Proceedings of 22nd Annual Symposium on Principles of distributed computing (PODC), 2003,* 63–72 (2003).

87. Kwon, A., Lazar, D., Devadas, S. & Ford, B. Riffle: An Efficient Communication System With Strong Anonymity. *Proceedings on Privacy Enhancing Technologies,* 115–134. ISSN: 2299-0984 (2016).

88. Larson, S. *Every single Yahoo account was hacked - 3 billion in all - Oct. 3, 2017* Oct. 2017. <http://money.cnn.com/2017/10/03/technology/business/yahoo-breach-3-billion-accounts/index.html> (visited on 02/12/2018).

89. Larson, S. *Fitness app that revealed military bases highlights bigger privacy issues* Jan. 2018. <http://money.cnn.com/2018/01/29/technology/strava-privacy-data-exposed/index.html> (visited on 02/12/2018).

90. Leskovec, J., Kleinberg, J. & Faloutsos, C. *Graphs over time* in *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05* **11** (ACM, New York, NY, USA, 2005), 177. ISBN: 159593135X. doi:10.1145/1081870.1081893. <http://eprints.pascal-network.org/archive/00001220/%7B%%7D5Cnhttp://portal.acm.org/citation.cfm?doid=1081870.1081893>.

91. Li, N., Li, T. & Venkatasubramanian, S. *t-Closeness: Privacy beyond k-anonymity and l-diversity* in *Proceedings - International Conference on Data Engineering* (2007), 106–115. ISBN: 1424408032. doi:10.1109/ICDE.2007.367856.

92. Li, X.-Y., Wan, P.-J. & Frieder, O. Coverage in wireless ad hoc sensor networks. *IEEE Transactions on Computers* **52,** 753–763. ISSN: 0018-9340 (2003).

93. Li, X.-Y., Wan, P.-J. & Wang, Y. Power efficient and sparse spanner for wireless ad hoc networks. *Proceedings Tenth International Conference on Computer Communications and Networks* **00,** 564–567. ISSN: 1095-2055 (2001).

94. Lin, D., Sherr, M. & Loo, B. T. *Scalable and Anonymous Group Communication with MTor* in *Proceedings on Privacy Enhancing Technologies* (2015). doi:https://doi.org/10.1515/popets-2016-0003.

95. Liptak, A. *Strava's fitness tracker heat map reveals the location of military bases* Jan. 2018. (Visited on 02/12/2018).

96. M. Dorigo and G. Dicaro. in *McGraw Hill, London* (eds Corne, D., Dorigo, M., Glover, F., Dasgupta, D., Moscato, P., Poli, R. & Price, K. V.) 11–32 (McGraw-Hill Ltd., UK, Maidenhead, UK, England, 1999). ISBN: 0-07-709506-5. <http://dl.acm.org/citation.cfm?id=329055.329062>.

97. MacAskill, E., Borger, J., Hopkins, N., Davies, N. & Ball, J. *GCHQ taps fibre-optic cables for secret access to world's communications* 2013. <https://www.theguardian.com/uk/2013/jun/21/gchq-cables-secret-world-communications-nsa> (visited on 03/27/2017).

98. Machanavajjhala, A., Kifer, D., Gehrke, J. & Venkitasubramaniam, M. l-Diversity. *ACM Transactions on Knowledge Discovery from Data* **1,** 3–es. ISSN: 15564681 (2007).

99. Matthews, W. & Coffrell, L. The PingER project: Active internet performance monitoring for the HENP community. *IEEE Communications Magazine* **38,** 130–136. ISSN: 01636804 (2000).

100. Meister, A. & Netzpolitik.org. *Das neue BND-Gesetz: Alles, was der BND macht, wird einfach legalisiert. Und sogar noch ausgeweitet.* June 2016. <https://netzpolitik.org/2016/das-neue-bnd-gesetz-alles-was-der-bnd-macht-wird-einfach-legalisiert-und-sogar-noch-ausgeweitet/> (visited on 04/24/2018).

101. Mendel, T., Puddephatt, A., Wagner, B., Hawtin, D. & Torres, N. *Global Survey on Internet Privacy And Freedom of Expression* ISBN: 978-92-3-104241-6. <http://unesdoc.unesco.org/images/0021/002182/218273e.pdf> (United Nations Educational, Scientific and Cultural Organization, 2012).

102. Menezes, A. J. (J. *et al. Handbook of Applied Cryptography* 5th ed., 816. ISBN: 0849385237. doi:10.1.1.99.2838 (CRC Press, 1996).

103. Merkle, R. C. Secure communications over insecure channels. *Communications of the ACM* **21,** 294–299. ISSN: 00010782 (1978).

104. Merkle, R. C. *Secrecy, Authentication, and Public Key Systems* PhD Thesis (Stanford, 1979).

105. Merkle, R. C. *One Way Hash Functions and DES* in *Advances in Cryptology - CRYPTO' 89 Proceedings* (Springer New York, New York, NY, 1989), 428–446. doi:10.1007/0-387-34805-0_40. <http://link.springer.com/10.1007/0-387-34805-0%7B%5C_%7D40>.

106. Mittal, P., Wright, M. & Borisov, N. *Pisces: Anonymous Communication Using Social Networks* in *Network and Distributed System Security Symposium (NDSS)* (2013). arXiv: 1208.6326. <papers3://publication/uuid/D1DA2EC4-06C5-4F02-A7B9-B540C756CCA5>.

107. Möller, U., Cottrell, L., Palfrader, P. & Sassaman, L. *Mixmaster Protocol - Version 2* tech. rep. (2003). <https://www.freehaven.net/anonbib/cache/mixmaster-spec.txt>.

108. Murdoch, S. J. & Danezis, G. *Low-cost traffic analysis of Tor* in *Proceedings - IEEE Symposium on Security and Privacy* (IEEE Computer Society, Washington, DC, USA, 2005), 183–195. ISBN: 0769523390. doi:10.1109/SP.2005.12. <http://dx.doi.org/10.1109/SP.2005.12>.

109. Narayanan, A. & Shmatikov, V. *Robust de-anonymization of large sparse datasets* in *Proceedings - IEEE Symposium on Security and Privacy* (2008), 111–125. ISBN: 9780769531687. doi:10.1109/SP.2008.33. arXiv: 0610105v2 [arXiv:cs].

110. Newman, M. E. J. Models of the Small World. *Journal of Statistical Physics* **101,** 819–841. ISSN: 00224715 (2000).

111. Open Whisper Systems. *Private Group Messaging* 2014. <https://whispersystems.org/blog/private-groups/> (visited on 11/17/2017).

112. Open Whisper Systems. *WhatsApp's Signal Protocol integration is now complete* 2016. <https://whispersystems.org/blog/whatsapp-complete/> (visited on 11/17/2017).

113. Parameswaran, M., Susarla, a. & a.B. Whinston. P2P networking: an information sharing alternative. *Computer* **34,** 31–38. ISSN: 0018-9162 (2001).

114. Parekh, S. Prospects for remailers: Where is anonymity heading on the internet? *First Monday.* ISSN: 13960466 (1996).

115. Parliament, E. *Report on the existence of a global system for the interception of private and commercial communications (ECHELON interception system) (2001/2098(INI))* 2001. doi:A5-0264/2001. <http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//TEXT+PRESS+DN-20010905-1+0+DOC+XML+V0//EN%7B%5C#%7DSECTION1>.

116. Pfitzmann, A. & Köhntopp, M. in *Designing Privacy Enhancing Technologies* (Springer, 2001). ISBN: 9783540417248. doi:10.1007/3-540-44702-4_1.

117. Pigné, Y., Dutot, A., Guinand, F. & Olivier, D. GraphStream: A Tool for bridging the gap between Complex Systems and Dynamic Graphs. *CoRR* **abs/0803.2.** arXiv: 0803.2093. <http://arxiv.org/abs/0803.2093> (2008).

118. Piotrowska, A., Hayes, J., Elahi, T., Meiser, S. & Danezis, G. *The Loopix Anonymity System* in *Usenix Security Symposium* (Vancouver, 2017). arXiv: 1703.00536. <http://arxiv.org/abs/1703.00536>.

119. Preneel, B. Cryptographic hash functions. *European Transactions on Telecommunications* **5,** 431–448. ISSN: 15418251 (1994).

120. Quinn, B. & Arthur, C. *PlayStation Network hackers access data of 77 million users* Apr. 2011. <https://www.theguardian.com/technology/2011/apr/26/playstation-network-hackers-data> (visited on 02/12/2018).

121. Rabin, M. O. *Digitalized Signatures and Public Key Functions as Intractable as Factorization* tech. rep. (Massachusetts Institute of Technology, Cambridge, 1979).

122. Raiciu, C. & Rosenblum, D. S. *Enabling confidentiality in Content-Based Publish/Subscribe infrastructures* in *2006 Securecomm and Workshops* (IEEE, Baltimore, USA, Aug. 2006), 1–11. ISBN: 1424404231. doi:10.1109/SECCOMW.2006.359552. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4198812>.

123. Raymond, J.-F. Traffic analysis: Protocols, attacks, design issues, and open problems. *Designing Privacy Enhancing Technologies,* 10–29. ISSN: 16113349 (2001).

124. Reed, M., Syverson, P. F. & Goldschlag, D. *Anonymous connection and onion routing* in *1997 IEEE Symposium on Security and Privacy* **16** (1998), 482, 494. ISBN: 0-8186-7828-3. doi:10.1109/49.668972. <http://dl.acm.org/citation.cfm?id=882493.884368>.

125. Reiter, M. K. M. & Rubin, A. D. A. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security (TISSEC),* 66–92. ISSN: 10949224 (1998).

126. Rohrer, J. *No Title* 2006. <http://mute-net.sourceforge.net/technicalDetails.shtml>.

127. Rossi, R. *ANts P2P* 2004. <http://antsp2p.sourceforge.net/>.

128. Samarati, P. & Sweeney, L. Protecting Privacy when Disclosing Information: k-Anonymity and its Enforcement Through Generalization and Suppresion. *IEEE Symposium on Research in Security and Privacy,* 384–393 (1998).

129. Schiavoni, V., Rivière, E. & Felber, P. *WHISPER: Middleware for confidential communication in large-scale networks* in *Proceedings - International Conference on Distributed Computing Systems* (IEEE, 2011), 456–466. ISBN: 9780769543642. doi:10.1109/ICDCS.2011.15.

130. Serjantov, A. & Danezis, G. in, 41–53 (Springer, Berlin, Heidelberg, 2003). ISBN: 3-540-00565-X. doi:10.1007/3-540-36467-6_4. <http://link.springer.com/10.1007/3-540-36467-6%7B%5C_%7D4>.

131. Shannon, C. E. A mathematical theory of communication. *The Bell System Technical Journal* **27,** 379–423. ISSN: 07246811 (1948).

132. Shannon, C. E. Communication theory of secrecy system. *The Bell System Technical Journal* **28,** 656–715 (1949).

133. Sherwood, R., Bhattacharjee, B. & Srinivasan, A. *P5 : a protocol for scalable anonymous communication* in *IEEE Symposium on Security and Privacy (S&P)* (IEEE, 2002), 58–70. ISBN: 0-7695-1543-6. doi:10.1109/SECPRI.2002.1004362.

134. Shikfa, A., Önen, M. & Molva, R. *Privacy and confidentiality in context-based and epidemic forwarding* in *Computer Communications* **33** (IEEE, 2010), 1493–1504. ISBN: 9781424444397. doi:10.1016/j.comcom.2010.04.035.

135. Shimbel, A. Structural parameters of communication networks. *The Bulletin of Mathematical Biophysics* **15,** 501–507. ISSN: 00074985 (Dec. 1953).

136. Stoica, I., Morris, R., Karger, D., Kaashoek, M. F., Balakrishnan, H., Stoica, I., Morris, R., Karger, D., Kaashoek, M. F. & Balakrishnan, H. *Chord: A scalable peer-to-peer lookup service for internet applications* in *ACM SIGCOMM Computer Communication Review* **31** (ACM, 2001), 149–160. ISBN: 1-58113-411-8. doi:10.1145/964723.383071.

137. Stutzbach, D., Rejaie, R., Duffield, N., Sen, S. & Willinger, W. *Sampling Techniques for Large, Dynamic Graphs* in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (Apr. 2006), 1–6. ISBN: 1-4244-0221-2. doi:10.1109/INFOCOM.2006.39.

138. Stutzbach, D., Rejaie, R., Duffield, N., Sen, S. & Willinger, W. *On unbiased sampling for unstructured peer-to-peer networks* in *IEEE/ACM Transactions on Networking* **17** (ACM, New York, NY, USA, 2009), 377–390. ISBN: 1-59593-561-4. doi:10.1109/TNET.2008.2001730. <http://doi.acm.org/10.1145/1177080.1177084>.

139. Sweeney, L. k-Anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **10,** 557–570. ISSN: 0218-4885 (2002).

140. Tanenbaum, A. S. & Wetherall, D. J. *Computer Networks* 5th ed. ISBN: 978-0-13-212695-3 (Pearson Education Inc., 2011).

141. Tariq, M. A., Koldehofe, B., Altaweel, A. & Rothermel, K. *Providing basic security mechanisms in broker-less publish / subscribe systems* in *4th ACM International Conference on Distributed Event-Based Systems (DEBS)* (ACM Press, 2010), 38–49. ISBN: 9781605589275. doi:10.1145/1827418.1827425.

142. *The Federation* June 2018. (Visited on 06/02/2018).

143. Troncoso, C., Gierlichs, B., Preneel, B. & Verbauwhede, I. *Perfect matching disclosure attacks* in *Lecture Notes in Computer Science* **5134 LNCS** (Springer, 2008), 2–23. ISBN: 3540706291. doi:10.1007/978-3-540-70630-4_2.

144. Tufekci, Z. & King, B. *We Can't Trust Uber* 2014. <https://www.nytimes.com/2014/12/08/opinion/we-cant-trust-uber.html>.

145. Valipour, S., Volk, F., Grube, T., Böck, L., Karg, L. & Mühlhäuser, M. *A formal holon model for operating future energy grids during blackouts* in *SMARTGREENS 2016 - Proceedings of the 5th International Conference on Smart Cities and Green ICT Systems* (2016), 146–153. ISBN: 9789897581847.

146. Van den Hooff, J., Lazar, D., Zaharia, M. & Zeldovich, N. *Vuvuzela: scalable private messaging resistant to traffic analysis* in *Proceedings of the 25th Symposium on Operating Systems Principles (SOSP)* (2015), 137–152. ISBN: 9781450338349. doi:10.1145/2815400.2815417.

147. Von Notz, K. & Ströbele, H.-C. *BND wird offiziell zur Massenüberwachungsmaschine* June 2016. <https://www.gruene-bundestag.de/presse/pressemitteilungen/2016/juni/bnd-wird-offiziell-zur-massenueberwachungsmaschine-28-06-2016.html> (visited on 04/24/2018).

148. Wang, C., Carzaniga, A., Evans, D. & Wolf, A. L. *Security issues and requirements for Internet-scale publish-subscribe systems* in *Proceedings of the Annual Hawaii International Conference on System Sciences* (IEEE, 2002), 3940–3947. ISBN: 0769514359. doi:10.1109/HICSS.2002.994531.

149. Wolinsky, D. I., Corrigan-gibbs, H., Ford, B. & Johnson, A. Dissent in Numbers : Making Strong Anonymity Scale. *USENIX Symposium on Operating Systems Design and Implementation (OSDI),* 179–192 (2012).

150. Xiao, X. & Tao, Y. m-Invariance: towards privacy preserving re-publication of dynamic datasets. *Proceedings of the 2007 ACM SIGMOD international conference on Management of data - SIGMOD '07,* 689–700. ISSN: 07308078 (2007).

151. Yao, Z., Leonard, D., Wang, X. & Loguinov, D. *Modeling heterogeneous user churn and local resilience of unstructured P2P networks* in *Proceedings - International Conference on Network Protocols, ICNP* (2006), 32–41. ISBN: 1424405939. doi:10.1109/ICNP.2006.320196.

152. Yao, Z., Wang, X., Leonard, D. & Loguinov, D. *On node isolation under churn in unstructured P2P networks with heavy-tailed lifetimes* in *Proceedings - IEEE INFOCOM* (2007), 2126–2134. ISBN: 1424410479. doi:10.1109/INFCOM.2007.246.

EHRENWÖRTLICHE ERKLÄRUNG

Hiermit erkläre ich, die vorgelegte Arbeit zur Erlangung des akademischen Grades *"Doctor rerum naturalium (Dr. rer. nat.)"* mit dem Titel

Efficient Anonymous Group Communication

selbständig und ausschließlich unter Verwendung der angegebenen Hilfsmittel erstellt zu haben. Ich habe bisher noch keinen Promotionsversuch unternommen.

Mühltal, Mai 2018

Tim Grube