Computer Networks 123 (2017) 51–63

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

# Explore and wait: A composite routing-delivery scheme for relative profile-casting in opportunistic networks

Carlos Borrego[a,*], Adrián Sánchez-Carmona[a], Zhiyuan Li[b], Sergi Robles[a]

[a] Department of Information and Communications Engineering Autonomous University of Barcelona Barcelona, Spain
[b] Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks Jiangsu, China

ARTICLE INFO

*Article history:*
Received 18 October 2016
Revised 8 May 2017
Accepted 8 May 2017
Available online 9 May 2017

*Keywords:*
OppNet
Bundle protocol
Profile-cast
Optimisation
DTN

ABSTRACT

In the context of Opportunistic Networking (OppNet), designing routing and delivery protocols is currently an open and active line of research. In some OppNet scenarios, destination addresses are not always known by sending applications. Profile-cast models solve this problem by allowing message destinations to be users or groups of users defined by their profiles. These profiles provide very effective ways of characterizing nodes in terms of node's attributes such as their profession, interests or typical whereabouts, for example. However, there are strong limitations in OppNet Profile-casting. There is no current way of representing special profiles defined by relative delivery functions such as *best, maximum* or *over-the-average*: nodes belong to these relative profiles taking into account not only attributes from the very same node but also relative to others from the same profile. In this article, we introduce Relcast, a Profile-cast model that allows messages to be sent to profiles defined in terms of relative delivery functions. Additionally, we present Explore and Wait, a composite routing-delivery scheme that uses optimal stopping theory-based delivery strategies to route Relcast messages. We show, using simulations, that this routing-delivery scheme performs better than traditional approaches that use state-of-the-art routing-delivery primitives.



## 1. Introduction

Opportunistic Networking (OppNet) [22] is a network paradigm where mobile nodes communicate with each other even if end-to-end connectivity among them never exists. In the context of OppNet, available destination addresses are not always known by the sending applications. Profile-based network models [13], where the destination of a message is not identified by a network/transport address, are very useful in OppNet. In this case, applications send messages to nodes belonging to a profile defined by one or more attributes rather than to specific addresses. For example, a user could send a message to another user belonging to a *neighbour profile*, defined as all users that live less than 1 km away from the current location.

However, there are strong limitations in OppNet Profile-casting. In particular, there is no current way of representing profiles defined by delivery functions such as *best, maximum, over-the-average* or *k-best*. We call these profiles relative profiles: nodes belong to these relative profiles taking into account not only attributes from the very same node but also relative to others from the same profile. For instance, following the previous example, a user application might want to send a message to the user belonging to the *neighbour* profile that has visited a certain place more times than anyone else from this profile.[1]

Nonetheless, this relative delivery decision is a complex task to perform. The reason for this is that intermittently connected OppNet nodes do not have access to the state of the network with regard to the attributes that define these relative profiles. Broadcasting messages to query the state of these attributes can be inaccurate and slow because of the characteristics of the network. Instead, sending single messages to the network to explore the network and afterwards deliver the message will imply a new problem: the problem of deciding when to stop exploring the network.

In this article, we overcome these limitations by allowing messages to carry state variables to be able to implement complex delivery strategies. These strategies are based on optimal stopping theory, a statistical tool for analysing the problem of choosing an optimal time to take a particular action. We introduce Relcast, a

* Corresponding author.
*E-mail addresses:* cborrego@deic.uab.cat (C. Borrego), adria.sanchez@deic.uab.cat (A. Sánchez-Carmona), lizhiyuan81@126.com (Z. Li), sergi.robles@uab.cat (S. Robles).

[1] In early versions of the Foursquare social network, if a user had checked into a certain place more days than anyone else, he/she would become *Mayor* of that place.

http://dx.doi.org/10.1016/j.comnet.2017.05.007
1389-1286/

profile-based relative network model to allow messages to be sent to profiles using relative criteria. Additionally, we present, Explore and Wait, a composite routing-delivery scheme based on optimal stopping theory to optimise the delivery of Relcast messages.

Like other protocols as Spray and Wait [27], Explore and Wait operates in two phases. During the first phase, the Explore phase, the message studies the network and understands about the nodes attributes by being forwarded to as many nodes from this profile as possible. We propose using a profile-based probabilistic routing protocol to facilitate these contacts: messages are routed in terms of historical profile encounters. During the second phase, the Wait phase, the message has acquired enough knowledge about the network, and it is ready to be delivered. The message waits, while still being routed until the delivery strategy considers the message should be delivered to the forwarded node. The result is a general purpose network-layer scheme that provides OppNet applications with the possibility of implementing relative delivery strategies.

Finally, we present different useful applications and a wide set of simulations in two different scenarios: an urban and a rural scenario. We base our simulations on Active-DTN [2,3], an implementation of the Bundle Protocol that allows bundles to have an active routing and delivery behaviour. Active-DTN allows applications to send Relcast bundles by including within the bundle their delivery protocols as bundle extensions implemented with mobile code. The results of these simulations show that our proposal performs better than using traditional network primitives since Relcast allows new group communication behaviours that are capable of being addressed using optimal strategies.

The rest of the paper is organised as follows. Section 2 reviews the state of the art. Section 3 introduces the Relcast network model and the Explore and Wait protocol. In Section 4, the experimentation using simulations is described. Finally, in Section 5, results and the conclusions that are drawn are presented.

## 2. Background

In this section, we study the state of the art of three different OppNet issues directly related to the proposal of this study: Profile-casting, mobile code applied to OppNet and stateful routing. Additionally, we present a list of motivating scenarios that can take advantage of our proposal.

### 2.1. Related work

#### 2.1.1. Profile-casting in OppNet

Opportunistic Networking (OppNet) is a network paradigm where nodes suffer from intermittent contacts, long latencies, low delivery ratios and indeterminate mobility patterns. Messages in OppNet are routed from their source to their destination in an opportunistic way using intermediate nodes.

There have been some interesting proposals for profile-based delivery in OppNet. In [13], the authors present a novel delivery approach where messages are sent to a sub-group of users as defined by their profiles (e.g. interests, social affiliation, etc.). They study large data sets of user mobility profiles and present a case-study of mobility Profile-cast with a similarity-based forwarding protocol. The same authors have published a similar work [14] where a new behaviour-oriented communication paradigm in mobile networks named Profile-cast, motivated by tight user-network coupling in mobile societies, is presented. In this novel paradigm, the authors proposed that messages are sent to sender-specified target profiles, instead of machine IDs. They present a systematic framework for such services. Additionally, they present a communication protocol in mobile networks based on the stability of the user behavioural profile to discover the receivers implicitly, abbreviated as CSI. In this protocol, messages are forwarded to a node if this node has a similarity to the target profile higher than the one of the custodian of the message.

Identifying users according to their affiliation, as first presented by Hui and Crowcroft [15], can improve forwarding decisions. In [8], the authors propose an efficient social profile-based routing scheme. The efficiency of this proposed scheme has been confirmed by trace-driven simulation, which also reflects the efficacy of exploring social features in OppNet. In [21], the authors present Peoplerank, a forwarding approach similar to PageRank where nodes are ranked using weighted social information. In a similar way, the authors of [33] introduce Predict and Relay, a routing algorithm based on foreseeability and semi-deterministic node's movement models.

Profile-cast seems to be a very interesting network model in OppNet. However, traditional delivery strategies are limited to node attributes. When facing more complex delivery strategies like, for example, relative-based ones, such as sending messages to the best node from a given profile, these proposals fail to give good general and dynamic network-layer solutions. As a consequence of this, some applications are not possible in OppNet.

#### 2.1.2. Stateful routing in OppNet

There are OppNet routing protocols that require keeping the internal state of the message with the purpose of remembering preceding events. For example, in [27], the popular Spray and Wait routing protocol *sprays* a number of copies into the network, and then *waits* until one of these nodes meets the destination. The number of times a message can be replicated is proposed to be included in the message, and decreases every time the message is *sprayed*. This is a very good example of message state. Unfortunately, these message states are not able to be implemented in the Bundle Protocol: there is no current proposal for including such state in the Bundle Protocol messages.

Mobile code along with message state has proven to be a good combination in a variety of settings. In [3] software code and message state are used to improve OppNet performance. The authors provide a solution that extends the messages being communicated by incorporating software code for forwarding, delivery, lifetime control, and prioritisation purposes. The proposal stems from the idea of moving the routing and the delivery algorithms from the router to the message.

Finally, very recently, proposals like [17] have started to apply to OppNet optimal stopping theories [5,23], statistical solutions for the problem of choosing a moment to make a particular decision to maximise a certain reward. This statistical approach has been applied to network issues such as routing and node searching.

### 2.2. Motivating scenarios

In this section, we discuss some motivating scenarios where traditional Profile-cast models fail to give efficient solutions.

#### 2.2.1. Urban OppNet

As explained in studies like [28], Internet service providers (ISPs) are using package inspection techniques to read and store users' messages and personal data. The authors of this study explain that these technologies are progressively gaining legal legitimacy. Additionally, cloud services can ban users when connecting using Tor-like anonymous services.[2]

As a consequence of this, OppNet has become a network paradigm that can be used as an alternative to traditional Internet infrastructure-based networks in order to preserve private

[2] A list of services blocking TOR: https://www.trac.torproject.org/projects/tor/wiki/org/doc/ListOfServicesBlockingTor.