Robust fully distributed file caching for delay-tolerant networks: A reward-based incentive mechanism

Sidi Ahmed Ezzahidi¹, Essaid Sabir², Sara Koulali³ and El-Houssine Bouyakhf¹

Abstract
This article exhibits a reward-based incentive mechanism for file caching in delay-tolerant networks. In delay-tolerant networks, nodes use relay’s store-carry-forward paradigm to reach the final destination. Thereby, relay nodes may store data in their buffer and carry it till an appropriate contact opportunity with destination arises. However, the relays are not always willing to assist data forwarding due to a limited energy or a low storage capacity. Our proposal suggests a reward mechanism to uphold and to sustain cooperation among relay nodes. We model this distributed network interaction as a non-cooperative game. Namely, the source node offers to the relay nodes a positive reward if they accept to cache and to forward a given file successfully to a target destination, whereas the relay nodes may either accept or reject the source deal, depending on the reward attractiveness and on their battery status (their actual energy level). Next, full characterizations of both pure and mixed Nash equilibria are provided. Then, we propose three fully distributed algorithms to ensure convergence to the Nash equilibria (for both pure equilibrium and mixed equilibrium). Finally, we validate our proposal through extensive numerical examples and many learning simulations and draw some conclusions and insightful remarks.

Keywords
Delay-tolerant network, content caching, energy efficiency, reward-penalty incentive mechanism, Nash equilibrium, distributed learning

Date received: 19 September 2016; accepted: 22 February 2017

Academic Editor: Sergio Toral

Introduction
Communication networks, whether they are wired or wireless, require that the end-to-end link always exists between a sender and a recipient. However, some emerging networks such as delay-tolerant networks (DTNs)¹ introduce new concepts like intermittent connectivity and large delay-tolerant services. Yet, DTNs are complex and distributed systems where a permanent link between pair nodes source/destination may only exist for brief and unpredictable periods of time due to the lack of network infrastructures. Nevertheless, the large delay is tolerated in such a network and the store-carry-forward paradigm of relays is designed to make a full end-to-end transmission possible. The DTN relays connect ubiquitously over a direct link instead of passing through a base station in order to provide an enhanced coverage, high data rate, low latency, and low cost.

¹LIMIARF Laboratory, FSR, Mohammed V University, Rabat, Morocco
²NEST Research Group, LRI Lab, ENSEM, University of Hassan II Casablanca, Casablanca, Morocco
³MATSI Laboratory, Mohammed I University, Oujda, Morocco

Corresponding author:
Essaid Sabir, NEST Research Group, LRI Lab, ENSEM, University of Hassan II Casablanca, Casablanca 20000, Morocco.
Email: e.sabir@ensem.ac.ma
Consequently, DTN paradigms may improve the resource utilization and may reduce the load of base stations. Furthermore, higher energy efficiency and enhanced network availability/robustness can be achieved. Such a network provides modular solutions to assist data transmission within intermittent forwarding where steady connectivity is hard or even impossible to establish. From a practical perspective, the DTN concept has been extensively used in several fields. For instance, it has been deployed to strengthen military communications, underwater networks, Vehicular Ad hoc NETworks (VANETs), and many other applications.

In a DTN context, when the source–relay encounter takes place (i.e. they are into communication range of each other), the source transmits its content files to the relay. This latter stores/caches them, it waits arbitrarily until it experiences a link opportunity and forwards them to the target destination. Therefore, the relay nodes must be able to buffer files for long periods of time. Indeed, these relays can be wireless devices with limited battery lifetime and storage capability. Hence, they may not be always available to assist the file transmission, which makes connectivity intermittent and arbitrary. Thus, designing an efficient and effective scheme is crucial. In this article, we build a rewarding mechanism encouraging the relay nodes to participate in the forwarding game. We believe such a scheme makes sense in DTN and can improve the overall network performance metrics.

Literature review

During the last few years, there has been a tremendous growing interest in DTN. Most of related current research focuses on improving the DTNs performance (e.g. routing/forwarding, scheduling policy, buffer management, and mobility models). More precisely, these works mostly focus on reducing (1) the messages loss probability, (2) the end-to-end delay, (3) the expected number of transmissions, and (4) the energy consumption while maximizing the delivery rate of transmitted messages. However, most of these provided results are based on some strong assumptions and/or are sometimes difficult to apply to a realistic network.

In the meantime, numerous research works have been devoted to stimulate DTN relays to participate in the relaying of data. This research direction has gained considerable interest and has become a hot topic. Indeed, several incentive mechanisms have been designed to sustain cooperation among selfish relay nodes. Jiang and Bai6 exhibited an interesting survey on incentive mechanisms for DTNs. Four schemes have been compared as follows: (1) virtual currency-based incentive mechanism, (2) credit-based incentive mechanism, (3) game-theory-based incentive mechanism, and (4) combined incentive mechanism. In the work by Hulke and Attar,7 the authors discussed different game theoretical-inspired incentive mechanisms and analyzed them while pointing out their advantages and drawbacks. Chahin et al.8 propose a fixed rewarding mechanism. To make tradeoff between incentives for cooperation and energy consumption, a Minority Game is constructed. Next, a learning algorithm is suggested to drive the system to a stable operating point (Nash equilibrium). In the work by El-Azouzi et al.,9 the authors use evolutionary game theory and a fixed rewarding mechanism to model the behavior of the relay nodes in a dense DTN environment. In the work by Brun et al.,10 the authors consider a simple reward scheme and provide some interesting insights on how the source node could optimize the reward value based on received relays information. This incentive mechanism depends on the time at which the source meeting occurs and on the information hidden/disclosed by the source (number of message copies/previous meetings with opponent, etc.), and both static and dynamic policies have been investigated. Lu et al.11 present how energy harvesting can be exploited to improve the performance of opportunistic forwarding in mobile DTNs. In the work by Niyato et al.,12 the authors discuss how the relay nodes behave and form coalitions to help each other to forward a given file while dealing with energy and buffer space constraints. Neglia and Zhang13 propose an activation forwarding control at relay nodes, but their optimization supposes that relays are always available to cooperate with the source node. However, the relays cannot always be active due to limited energy or buffer space.

Another exciting research direction focuses on buffer management and content popularity. Yet, several studies mainly spotlight the buffer management, for example, Krifa et al.14 propose an optimal buffer management policy under unlimited bandwidth, which is far from reality. Wang et al.,15 consider that the popularity of the content files is a crucial parameter to distribute them on relay’s buffer. Next, they propose an algorithm that discovers a near-optimal allocation of files in the network. This work uses a minor unrealistic assumption that the files have the same size, and a major unrealistic assumption of a dense network is guaranteeing file transmission to the destination. In the work by Koulaí et al.,16 the authors propose a strategic beaconing scheme for DTN and suggest an application for unmanned vehicle networks (UAVs). The main idea is to strategically switch to sleep mode when a contact opportunity may only occur with very low probability. This mechanism has been shown to meet a delivery-energy tradeoff. In the work by Le et al.,17 the authors propose a cooperative caching scheme based on the social relationship among nodes to improve the DTNs performance metrics. More precisely, this may
help reduce the delay and the cost of message replicas. Yin and Cao\textsuperscript{18} design a cooperative caching scheme for mobile ad hoc networks (MANETs), aiming to reduce the delay of future requests by caching the data at certain relay nodes. Numerous other works deal with the reduction of the data redundancy among neighboring nodes to create content diversity with a limited cache space.\textsuperscript{19,20} However, these caching schemes are hard to implement due to the intrinsic challenging characteristics of DTNs.

**Our contribution**

In this work, we introduce a mechanism for file caching in DTNs through sustaining cooperation among relay nodes by offering them a reward. We evaluate the tradeoff between the reward value and the energy consumed to find out a rest/stable point for the distributed network. The mechanism consists of asking the relays to cache a file according to an incentive rewarding. More precisely, the source node is offering a reward to the first relay node for accepting to cache the file and succeeding to forward it to the final destination. However, the relay may either accept or decline the source offer depending on the reward attractiveness and on its battery status (energy cost). For computation tractability and without loss of generality, we consider the two-hop routing\textsuperscript{21} to route the files to their destinations. Moreover, the choice of the two-hop routing is also motivated by an efficient resources management, that is, bandwidth, buffer space, and relaying energy. The source node attempts to transmit the file to any relay that it encounters, whereas the relay nodes are allowed to transmit the file only to the final destination.

Due to limited storage and battery constraints, relay nodes may behave in a selfish way, that is, they may not be willing to cooperate all the time. Hence, the interaction among the source node and the relay nodes is naturally modeled as a non-cooperative game. The dynamic interaction among relays and the source node (all assumed to be rational) pushes the system to converge to an equilibrium point.\textsuperscript{22} Now, each player seeks independently to maximize its own utility function which is also depending on the other players strategy vector. It follows that Nash equilibrium (Nash equilibrium is a strategy profile where no player/agent has incentive to unilaterally deviate) as a concept solution is the natural solution concept for such an adversarial situation. We define the payoff of each player a function of the contact probability (with the source node), the delivery probability (to the destination node), the file lifetime, and the energy consumed. On one hand, the source sets a reward value and invites relay nodes to accept its deal. On the other hand, each relay has two actions (pure strategies), either to accept (strategy “a”) or to reject (strategy “r”) the source’s offer. Next, we allow the relay nodes to mix their strategies according to some probability distribution (mixed strategy). In order to reach a delivery-energy tradeoff, we consider that only the first relay that encounters the destination within the file lifetime still receives the reward. Thus, the other nodes having accepted the deal will receive zero and incur an energy penalty. To sum up, the relay has a decision-making problem which consists of acting under constraint of its battery energy, the reward value, and the probability that another relay had delivered the transmitted file prior to it.

A preliminary version of this work\textsuperscript{23} presents the caching problem for two relay nodes. This scheme is quite unrealistic but still provides some useful insights toward the derivation of the general model. The work presented by Ezzahidi et al.\textsuperscript{24} provides a discussion on the reward mechanism for the n-person game but without detailed theoretic analysis and does not consider some important parameters. The major contributions of this article are fourfold and can be summarized as follows:

- Our scheme is realistic. Indeed, decision-making includes the contact probability, the file lifetime, the energy consumption, and the relay willingness to cooperate;
- We designed numerous fully distributed algorithms to reach the game equilibria: our algorithms cover both pure Nash equilibrium and mixed Nash equilibrium, for both discrete action sets (for relay nodes) and continuous action sets (for source node);
- Our proposed learning algorithms require only local information (no external information is required), and thus, our scheme has good scalability features. In other words, our framework is suitable for both sparse and dense/ultra-dense networks. It could also capture device-to-device-like communications.
- We showed that the Nash equilibrium of this game has some good fairness features. Indeed, we showed that the Price of Anarchy (PoA) is bounded and could be efficiently controlled and make it as closer to 1 as one wish by fine-tuning the file lifetime or alternatively adapting the mobility parameters (speed, directions, stop points, etc.).

The rest of this article is organized as follows. In section “System architecture and model formulation,” we describe the problem, system architecture, and the problem formulation. We analyze the game equilibria for two-person case and then for the n-person case in sections “Two-person caching game” and “The n-person caching game,” respectively. Next, we describe three learning schemes in section “Learning algorithms
and insights for real-world implementation.” Extensive numerical and simulation results are presented in section “Numerical investigations.” Many insightful concluding remarks and perspectives are drawn in section “Conclusion.”

**System architecture and model formulation**

We consider a DTN network with a single source, a single destination, and \( n \) relays which are involved in the transmission of files. The files are generated at the source and the relay nodes should forward them to their final destination. Each file has a utility time \( h \) (finite horizon) during which the destination is interested in such a content. We assume that the relays have sufficient buffer capacity to store the content file, and they are only authorized to forward it to the destination (essence idea of the two-hop routing). Moreover, we suppose that the contact time is quite sufficient to succeed the file-carrying transmission when the link between nodes is up and the inter-contact times between any pair of nodes are independent and identically distributed (i.i.d.) random variables. Furthermore, the source node rewards the relay nodes for agreeing to cache-and-forward the file. However, only the first relay that has accepted to cache the file and that has succeeded to deliver it to the target destination receives the reward (each file has its own reward). Of course, each participation in the caching game incurs a cost that can be seen as the energy consumed by reception, caching, and transmission operations. Due to mobility patterns, the contact among the nodes is intermittent. Several works (e.g. Gao et al.\(^{25}\)) showed that the inter-contact time between a pair of nodes follows an exponential distribution with rate \( \lambda \). Let \( p_c \) be the contact probability between two nodes within the file content lifetime \( h \). It is written as

\[
p_c = 1 - e^{-\lambda h}
\]

(1)

Throughout this article we assume, without loss of generality and clearness, that every node has the same contact probability with the source node. Besides, as mentioned above, at each encounter with the source, the relay chooses either to accept or to reject a given file. Hence, the source shall set an attractive payment strategy to encourage the relays to cooperate. It advertises a positive reward (The earned reward would be a certain credit, an amount of bit-coins, or some reputation bonus that the source uses to send its own files over the DTN network) \( \alpha \geq 0 \) and promises it for the first relay node accepting to cache and forward successfully the file to the target destination. Figure 1 illustrates the interaction between the source node, relay nodes, and the destination node.

**Source problem**

The behavior of the source can be captured from a business perspective, that is, when it generates a file, it associates with it a value \( \alpha \in [0, \alpha_{\text{max}}] \). The file value may reflect the importance of the file from the source perspective. Alternatively, this value may also be perceived as related to the freshness and the popularity of the content file. Due to the lack of simplicity, we consider that the reward to be advertised by the source node equals \( \alpha \). Thus, the source is seeking to minimize this value while ensuring that the relay nodes cooperate with it. When a source–relay contact takes place, if the relay accepts and successfully delivers the file to the
destination within deadline \( t \leq h \), the relay receives \( \alpha \) from the source node, and the source receives the surplus \( \alpha_{\text{max}} - \alpha \). However, when the relay accepts to cache the file but fails to deliver it to the target destination, the source node will receive penalty \( -\alpha_{\text{max}}\). The last case arises when no relay node accepts to cache the file, the source will be again charged \( -\alpha_{\text{max}}\). The last penalty aims to make incentive to guarantee a non-zero delivery rate.

**Relay problem**

When a file is generated by a source, a control admission takes place during the utility time \( h \). Yet, the relay has two possible strategies, either to accept (action “\( a \)”) or to reject (action “\( r \)”) this file. From strategic reasoning, the relay chooses to accept or not depending on the value \( \alpha \) and on the expected energy to be consumed during the relaying operation. Clearly, mutual strategies induce some payoff for both the relay node and the source node. More precisely, when a succeeded transmission occurs (i.e. the relay delivers the file to the destination within the file lifetime \( h \)), the relay receives a positive reward \( \alpha \) (the file value). However, whenever the relay accepts to cache the file but fails to forward it, it will experience a constant regret \( \beta \). We notice here that an energy cost \( \sigma_h \), which helps to increase the network lifetime, applies whenever the relay decides to participate in the game. When the relay picks out strategy “\( r \)”, it will either bear a cost \( \gamma \) if no contact with the destination occurs within \( h \), or it will bear a cost \( -\alpha \) if it encounters the destination within \( h \). The last penalty aims to sustain cooperation and then avoids always-reject strategy.

**Remark 1.** In a DTN environment, the contact opportunities are sparse. Then, each node (source, relay, and destination) must advertise periodic beacons to discover its neighboring nodes and then get aware about contact opportunities. In this article, we consider that this feature is used by every device, so it can be plausibly omitted, alternatively it can be added as a fixed additional energy cost. Thus, this will not change the obtained results since it only has a shifting effect.

Obviously, each transmission/reception attempt incurs an energy cost. We assume that each transmission consumes an amount of energy denoted by \( \sigma_r \), whereas the relay node may experience much higher energy consumption. Yet, the relay node utilizes its energy budget to receive the file from the source node, to store the file till it meets destination or file deadline, and to send the content file to the final destination. Let \( \sigma \) be the energy consumed during file storage per unit time. For simplicity and without loss of generality, we include the reception and the transmission energy costs into the forwarding energy denoted by \( \sigma_h \). Thus, we have

\[
\sigma_h = \int_0^h \sigma \lambda e^{-\lambda t} \, dt = \sigma \left[ 1 - (1 + \lambda h) e^{-\lambda h} \right] / \lambda
\]

It has been shown, in the works by Ezzahidi et al.\(^{24}\) and Altman,\(^{26}\) that a single relay node delivers a file to destination within the file lifetime \( h \) with probability \( 1 - Q_h \), where

\[
Q_h = (1 + \lambda h) e^{-\lambda h}
\]

is the probability that a relay is not succeeding in file relaying to destination. Moreover, we assume that the relays have the same success probability.

Thus, equation (2) becomes

\[
\sigma_h = \frac{\sigma}{\lambda} (1 - Q_h)
\]

Now, we deal with the utility functions of both the source node and the relay node under the interaction details and the reward described above. Besides, it is rational that any action played by relay depends on the actions tacked by \( n - 1 \) opponent relays, and thus, the utilities are also functions of the number of relays in the network.

Let us denote \( P_{\text{succ}}(\text{“}a\text{”}, z) \) the probability that a given relay \( i \) among \( z \) participants, played strategy “\( a \)” and succeeded to deliver a given file (i.e. the first to deliver the file to the destination). We have the following

\[
P_{\text{succ}}(\text{“}a\text{”}, z) = (1 - Q_h) \sum_{j=1}^{z} \left( \frac{z - 1}{j - 1} \right) (1 - Q_h)^{j} = 1 - Q_h^z
\]

We denote by \( n_{c} \) the number of relays that having accepted to cache the file out of \( \bar{n} \) relays encountered by the source node. The mean number of nodes encountered by the source equals

\[
\bar{n} = \sum_{k=0}^{n} \binom{\bar{n}}{k} k p_k (1 - p_c)^{n-k} = np_c = n \left( 1 - e^{-\lambda h} \right)
\]

**Relay utility**

The utility function of the relay is defined as the difference between the reward earned from the source node and the energy consumed during the cache-and-forward transaction. We denote by \( U_i(\text{“}a\text{”}, \alpha) \) the utility function of the relay when it plays its pure strategy accept “\( a \)” and \( U_i(\text{“}r\text{”}, \alpha) \) when it plays its pure strategy reject “\( r \)”

\[
\begin{align*}
U_i(\text{“}a\text{”}, \alpha) &= \alpha P_{\text{succ}}(\text{“}a\text{”}, n_{c}) - \beta (1 - P_{\text{succ}}(\text{“}a\text{”}, n_{c})) - \sigma_h \\
U_i(\text{“}r\text{”}, \alpha) &= -\alpha P_{\text{succ}}(\text{“}a\text{”}, n_{c}) - \gamma
\end{align*}
\]
Source utility

We define the utility function of the source node as the difference between the maximum value that it is willing to pay and the cost function (expected energy consumed). It is expressed by

\[
U_i(\alpha', \alpha) = n_i\alpha(\alpha_{\max} - \alpha)P_{\text{succ}}(\alpha', n_i) - n_i\alpha_{\max}[1 - P_{\text{succ}}(\alpha', n_i)] - (\bar{n} - n_i)(\alpha_{\max} - \alpha)
\]

and can be rewritten as

\[
U_i(\alpha', \alpha) = (\bar{n} - n_i)(P_{\text{succ}}(\alpha', n_i) + 1)\alpha - \bar{n}\sigma - \alpha_{\max}[\bar{n} - 2n_iP_{\text{succ}}(\alpha', n_i)]
\]

Now, we turn to study the node’s behavior and characterize the equilibria structure of the caching game. Here, the source and the relay nodes are rational, that is, they know all system parameters mentioned above and each one seeks to maximize its own objective function. More precisely, we will exhibit some nice properties of the Nash equilibria in terms of existence and uniqueness under both pure strategies and mixed strategies. Under these assumptions, the induced caching game is clearly a one-shot (static game) non-cooperative game with complete information and the corresponding results. This formalism is suitable for our problem since it allows to clearly model the node’s behavior at low complexity and acceptable computational tractability.

Existence of a Nash equilibrium for the game is given by the well-known Debreu-Fan-Glicksberg theorem. Yet, under continuity of each utility function \( U_i(p_i, p_j) \) in \( p_i \), the quasi-concavity of \( U_i \) in \( p_i \), and the convex-compactness of strategy sets, the uniqueness of equilibrium is guaranteed. Moreover, it can be achieved with the correspondence of the best response.

Definition 1. The strategy profile \( (p^*, \alpha^*) \) is a Nash equilibrium if it satisfies

1. \( \alpha^* \in \arg \max_{\alpha} U_i(\alpha, p^*) \), \( \alpha \in [0, \alpha_{\max}] \)
2. \( p^* \in \arg \max_{p} U_i(\alpha^*, p) \)

Hereafter, we derive some closed form for the system equilibria. We first consider the simple case of two-person game. Next, we generalize our results to the \( n \)-person game.

Two-person caching game

In this section, we present our results and characterize the Nash equilibria under pure strategies and mixed strategies.

Pure Nash equilibrium

When nodes are only using pure strategies, we have the following result.

Lemma 1. The strategy profile \( (\alpha', 0) \) is a pure Nash equilibrium if and only if

\[
h \geq -\frac{1}{\lambda} \left[1 + W\left[-\frac{\sigma - \lambda \gamma}{(\sigma - \lambda \beta)\bar{e}}\right]\right], \quad \text{and} \quad 0 < \frac{\sigma - \lambda \gamma}{\sigma - \lambda \beta} \leq 1,
\]

where \( W \) (Lambert-W function) is the reciprocal function of \( f(x) = xe^x \).

Proof. When the \( n \) relay plays the pure strategy \( \alpha' \), we have

\[
U_i(\alpha', \alpha) \geq U_i(\alpha', 0), \quad \forall i = 1, \ldots, n
\]

which means that

\[
\alpha(1 - Q_h) - \beta Q_h - \sigma_h \geq -\alpha(1 - Q_h) - \gamma
\]

In this case, the payoff of the source is written as

\[
U_i(\alpha', \alpha) = (\alpha_{\max} - \alpha)(1 - Q_h) - \alpha_{\max}Q_h - \sigma_h = \alpha_{\max}(1 - Q_h) - (1 - Q_h)\alpha - \sigma_h
\]

The source payoff is a linear function with a negative slope, and it reaches the maximum at \( \alpha^* = 0 \), with \( \alpha \in [0, \alpha_{\max}] \). Thus, by substituting \( \alpha^* = 0 \), equations (2) and (3) into equation (11), we obtain

\[
-\beta(1 + \lambda h)e^{-\lambda h} - \sigma \left[\frac{1 - (1 + \lambda h)e^{-\lambda h}}{\lambda}\right] \geq -\gamma
\]

After some algebra, we obtain

\[
h \geq -\frac{1}{\lambda} \left[1 + W\left[-\frac{\sigma - \lambda \gamma}{(\sigma - \lambda \beta)\bar{e}}\right]\right]
\]

which completes the proof.

Lemma 2. The strategy profile \( (\alpha', \alpha_{\max}) \) is a pure Nash equilibrium if and only if

\[
h \leq -\frac{1}{\lambda} \left[1 + W\left[-\frac{\sigma - \lambda(\gamma + 2\alpha_{\max})}{(\sigma - \lambda(\beta + 2\alpha_{\max}))\bar{e}}\right]\right]
\]

and

\[
0 < \frac{\sigma - \lambda(\gamma + 2\alpha_{\max})}{\sigma - \lambda(\beta + 2\alpha_{\max})} \leq 1
\]

Proof. Let us assume now that the relay nodes play the pure strategy \( \alpha' \) (reject). We have the following

\[
U_i(\alpha', \alpha) \geq U_i(\alpha', \alpha)
\]

Now, the payoff of the source node is
Again, the payoff function of the source node is a linear function but with a positive slope this time. Then, it reaches the maximum at $\alpha^* = \alpha_{\text{max}}$. By substituting $\alpha^* = \alpha_{\text{max}}$, equations (2) and (3) into equation (15), we obtain

$$-\alpha_{\text{max}}(1 - (1 + \lambda h)e^{-\lambda h}) - \gamma \geq \alpha_{\text{max}}(1 - (1 + \lambda h)e^{-\lambda h}) - \beta(1 + \lambda h)e^{-\lambda h} - \sigma\left[1 - (1 + \lambda h)e^{-\lambda h}\right]$$

After some algebra, we obtain

$$h \leq -\frac{1}{\lambda}\left(1 + \frac{\sigma - \lambda(\gamma + 2\alpha_{\text{max}})}{|\sigma - \lambda(\beta + 2\alpha_{\text{max}})|}\right)$$

which completes the proof.

Pure Nash equilibrium is a natural solution concept. However, such a solution may not exist all the time, see existence conditions in Lemma 1 and Lemma 2. Moreover, a pure equilibrium could fail to achieve a certain lucidity between relay nodes and good fairness properties among the game players. In fact, the pure Nash equilibrium ("a", 0) seems counter-intuitive since the relay cooperates though the value of reward is null. This can be explained by considering the content lifetime and the energy costs. When the value of $h$ is quite high, the probability $1 - Q_h$ that the relay will encounter the destination becomes high and then the relay node chooses often to cache the content. Here, a relay node will incur a high damage in case of declining to cache the content due to the fixed regret and the high encounter probability. For more energy efficiency and better fairness properties, we allow the game players to use mixed strategies. From now on, a Nash equilibrium point is a strategy profile where each relay accepts to cache the file with some probability, and none of them has incentive to unilaterally deviate.

**Mixed Nash equilibrium**

When mixed strategy is permitted, the relay node chooses its pure strategies according to some probability distribution. Now, it plays strategy "a" with probability $p \in [0, 1]$ and plays strategy "r" with probability $1 - p$.

Consider that the source node sets a file value $\alpha$ when the relay accepts the file with probability $p$. The relay will mix between its two pure strategies. At equilibrium, the expected utilities turn out to be equal, and the relay node becomes indifferent on which pure strategy to pick. Hence, the equilibrium will be given by solving the following equation (the indifference principle)

$$U_s(\text{"a"}, \alpha) = U_s(\text{"r"}, \alpha)$$

(17)

then

$$\alpha(1 - Q_h) - \beta Q_h - \sigma_h = -\alpha(1 - Q_h) - \gamma$$

(18)

which allows to compute the equilibrium content value to be set by the source node

$$\alpha^* = \frac{1}{2(1 - Q_h)}\left[\left(\frac{\beta}{\lambda}\right)Q_h + \frac{\sigma}{\lambda} - \gamma\right]$$

(19)

We turn now to derive the equilibrium forwarding strategy for relay nodes. The source problem can be resolved by maximizing its own objective function. At Nash equilibrium, the derivative with respect to $\alpha$ of its expected utility vanishes. Namely

$$\frac{\partial U_s(\text{"a"}, \alpha)}{\partial \alpha} = 0$$

(20)

The expected payoff of the source node writes

$$U_s(p, \alpha) = ((\alpha_{\text{max}} - \alpha)(1 - Q_h) - \alpha_{\text{max}}Q_h - \sigma) p$$

$$- (\alpha_{\text{max}} - \alpha)(1 - p)$$

(21)

then

$$\frac{\partial U_s(p, \alpha)}{\partial \alpha} = (Q_h - 2)p + 1 \Rightarrow p^* = \frac{1}{2 - Q_h}$$

Thus, the mixed Nash equilibrium is given by

$$\begin{cases}
\alpha^* = \frac{1}{2(1 - Q_h)}\left[\left(\frac{\beta}{\lambda}\right)Q_h + \frac{\sigma}{\lambda} - \gamma\right] \\
p^* = \frac{1}{2 - Q_h}
\end{cases}$$

(22)

**Remark 2** When the horizon (file lifetime) is very large, simple formulas for mixed Nash equilibrium $(\alpha^*, p^*)$ could be found. Namely

$$\begin{cases}
\lim_{h \to +\infty} \alpha^* = \frac{1}{2}\left(\frac{\sigma}{\lambda} - \gamma\right) \\
\lim_{h \to +\infty} p^* = \frac{1}{2}
\end{cases}$$

(23)

**PoA**

In this subsection, we aim to quantify the performance degradation caused by the selfish behavior of the non-cooperative relay nodes. The loss of efficiency due to decentralizing decision-making is often captured using the concept of PoA introduced by Koutsoupias and Papadimitriou. \textsuperscript{27} It is defined as the worst-achievable ratio between the aggregate utilities of a Nash equilibrium (decentralized setting) and of the optimal solution (centralized setting). When the value of the PoA is close to 1, it signifies that the difference between a Nash equilibrium and the optimal solution is not significant.
and near-optimal performance can be met without an expensive centralized control. However, unfortunately, the PoA can be arbitrarily large which means that the decentralized scheme may lead to a huge counter-performance and high loss of efficiency. Since the Nash equilibrium is unique for our caching game, the PoA can be written as follows

\[ \text{PoA} = \frac{U_{\text{opt}}(p_{\text{opt}}, a_{\text{opt}})}{U_{\text{eq}}(p^*, a^*)} \]  

(24)

Let us compute \( U_{\text{opt}}(p, a) \), the sum of the expected utility functions (social welfare), and derive its globally optimal solution. We have

\[
U_{\text{opt}}(p, a) = (\alpha (1 - Q_h) - \beta Q_h - \sigma_r) p \\
+ (\alpha (1 - Q_h) - \gamma (1 - p) + (\alpha_{\text{max}} - \alpha) (1 - Q_h) \\
- \alpha_{\text{max}} Q_h - \sigma_r) p - (\alpha_{\text{max}} - \alpha) (1 - p)
\]

After some algebra, we obtain

\[
U_{\text{opt}}(p, a) = \alpha Q_h - \alpha_{\text{max}} \\
\gamma \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - (\alpha + \beta) Q_h - \sigma_r + \gamma \right) p
\]

Lemma 3 The globally optimal solution of the function \( U_{\text{opt}}(p, a) \) occurs at \((a_{\text{opt}}, p_{\text{opt}})\) such that

\[
\begin{align*}
    p_{\text{opt}} &= 1 \\
    a_{\text{opt}} &= \frac{1}{Q_h} \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - \sigma_r + \gamma - \beta
\end{align*}
\]

Proof. It is quite straightforward by vanishing the first-order derivative of \( U_{\text{opt}}(p, a) \). Namely

\[
\begin{align*}
    \frac{\partial U_{\text{opt}}(p, a)}{\partial a} &= -Q_h p + Q_h = 0 \\
    \frac{\partial U_{\text{opt}}(p, a)}{\partial p} &= \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - (\alpha + \beta) Q_h \\
    -\sigma_r + \gamma &= 0
\end{align*}
\]

Thus

\[
\begin{align*}
    p_{\text{opt}} &= 1 \\
    a_{\text{opt}} &= \frac{1}{Q_h} \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - \sigma_r + \gamma - \beta
\end{align*}
\]

and

\[ U_{\text{opt}}(p_{\text{opt}}, a_{\text{opt}}) = \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - \sigma_r - \beta Q_h - \alpha_{\text{max}} \]  

(25)

Let us compute \( U_{\text{eq}}(p^*, a^*) \), the sum of the expected utility functions at Nash equilibrium. We have

\[
U_{\text{eq}}(p^*, a^*) = \alpha^* Q_h - \alpha_{\text{max}} - \gamma \\
+ \left[ \left( 2\alpha_{\text{max}} - \frac{\sigma}{\lambda} \right) (1 - Q_h) - (\alpha + \beta) Q_h - \sigma_r + \gamma \right] p^*
\]

(26)

Substituting equations (25) and (26) into equation (24) yields

\[
\text{PoA} = \frac{A(1 - Q_h) - \sigma_r - \beta Q_h - \alpha_{\text{max}}}{|A(1 - Q_h) - (\alpha + \beta) Q_h - \sigma_r + \gamma| p^* + B}
\]

with \( A = (2\alpha_{\text{max}} - \sigma/\lambda) \) and \( B = \alpha^* Q_h - \alpha_{\text{max}} - \gamma \).

We finally obtain

\[
\lim_{h \to +\infty} \text{PoA} = \frac{2[\sigma + \lambda(\sigma_r - \alpha_{\text{max}})]}{\sigma + \lambda(\sigma_r + \gamma)}
\]

(27)

The last result shows that the PoA of the caching game is bounded for content files with large lifetime. Yet, the PoA can be efficiently controlled by fine-tuning the source intrinsic parameters such as \( \alpha_{\text{max}} \). Otherwise, it can also adjust the relay nodes mobility pattern, that is, \( \lambda \), to control the loss of efficiency due to decision-making decentralizing.

The n-person caching game

This section extends our results of the general case of \( n \) relays, a single source, and a single destination. We analyze the interaction/competition among relay nodes and derive some sufficient conditions for existence of the Nash equilibria.

A strategy profile of the \( n \)-person caching game is a strategy vector \((A_n, R_{\tilde{n}-n_a}, a)\) where

1. \( A_n = ("a", "a", \ldots, "a") \) is the profile that \( n_a \) relays play strategy "a," that is, participate in the caching game;
2. \( R_{\tilde{n}-n_a} = ("r", "r", \ldots, "r") \) is the profile that \( \tilde{n} - n_a \) play strategy "r," decline to cache the file content;
3. The source node plays strategy \( a \).

Pure Nash equilibrium

Lemma 4. The strategy profile \((A_n, R_0, 0)\), (respectively \((A_0, R_n, \alpha_{\text{max}})\)) is a pure Nash equilibrium if

\[
\sigma_h \leq \gamma - \beta (1 - P_{\text{succ}}("a", \tilde{n}))
\]

(respectively \( \sigma_h \geq \gamma - \beta + (2\alpha_{\text{max}} + \beta) P_{\text{succ}}("a", \tilde{n}) \))

Proof. In the case where all relay nodes \( \tilde{n} \) play their strategy "a," we write
\[ U_i(\text{"}'a', \alpha) \geq U_i(\text{"}'r', \alpha), \quad \forall i \in \hat{n} \]

where \( P_{\text{suc}}(\text{"}'a', \hat{n}) - \beta(1 - P_{\text{suc}}(\text{"}'a', \hat{n})) - \sigma_h \geq \gamma \) (28)

Since \( n_a = \hat{n} \), the source’s utility becomes

\[ U_i(\text{"}'a', \alpha) = \hat{n} \alpha - \alpha \alpha_{\text{max}}(1 - 2P_{\text{suc}}(\text{"}'a', \hat{n})) - \hat{n} \sigma_t \]

which is a linear function with a negative slope. Then, it reaches its maximum at \( \alpha = 0 \).

Substituting \( \alpha = 0 \) into equation (28) yields

\[ - \beta(1 - P_{\text{suc}}(\text{"}'a', \hat{n})) - \sigma_h \geq - \gamma \] (29)

which completes the first part of the lemma. In the case where all relays \( \hat{n} \) decline to cache the file, that is, they play strategy \( "r", "n" \) we have

\[ U_i(\text{"}'r', \alpha) \geq U_i(\text{"}'a', \alpha), \quad \forall i \in \hat{n} \]

That is

\[ - \alpha P_{\text{suc}}(\text{"}'a', \hat{n}) - \gamma \geq \alpha P_{\text{suc}}(\text{"}'a', \hat{n}) - \beta(1 - P_{\text{suc}}(\text{"}'a', \hat{n})) - \sigma_h \] (31)

Since \( n_a = 0 \), the source node’s utility becomes

\[ U_i(\text{"}'a', \alpha) = \hat{n} \alpha - \alpha \alpha_{\text{max}} - \hat{n} \sigma_t \]

which is again a linear function with positive slope. Thus, it reaches its maximum at \( \alpha = \alpha_{\text{max}} \).

Substituting \( \alpha = \alpha_{\text{max}} \) into equation (31) yields

\[ - \alpha_{\text{max}} P_{\text{suc}}(\text{"}'a', \hat{n}) - \gamma \geq \alpha_{\text{max}} P_{\text{suc}}(\text{"}'a', \hat{n}) - \beta(1 - P_{\text{suc}}(\text{"}'a', \hat{n})) - \sigma_h \] (32)

\[ \Rightarrow \sigma_h \geq \gamma - \beta (2 \alpha_{\text{max}} + \beta) P_{\text{suc}}(\text{"}'a', \hat{n}) \]

The proof of the second part of the lemma is complete.

The next lemma provides sufficient conditions for existence of a pure Nash equilibrium where some relay nodes accept the source deal, whereas the remaining relay nodes decline the offer.

**Lemma 5.** The strategy profile \((A_n, R_{\hat{n} - n_a}, 0)\), (resp. \((A_n, R_{\hat{n} - n_a}, \alpha_{\text{max}})\)) is a pure Nash equilibrium if

\[
\begin{align*}
n_a & \geq \frac{\lambda \beta \hat{n}}{\lambda (2 \beta + \gamma) + \sigma (1 - Q_h)} \\
\text{(resp. } & \quad n_a \leq \frac{\lambda (2 \alpha_{\text{max}} + \beta) \hat{n}}{\lambda (2 \beta + 2 \alpha_{\text{max}} - \gamma) + \sigma (1 - Q_h)})
\end{align*}
\]

**Proof.** Let us denote by \( A \) the group of relay nodes that have accepted to cache-and-forward the file, with cardinality \(|\mathcal{A}| = n_a\). The group of relay nodes that have declined the source offer is denoted by \( \mathcal{R} \), with cardinality \(|\mathcal{R}| = \hat{n} - n_a\).

We have the following for cooperating relay nodes

\[ U_i(\text{"}'a', \alpha) \geq U_i(\text{"}'r', \alpha), \quad \forall i \in \mathcal{A} \] (33)

Similarly, we have the following for defecting relay nodes

\[ U_i(\text{"}'a', \alpha) \leq U_i(\text{"}'r', \alpha), \quad \forall i \in \mathcal{R} \] (34)

From equations (33) and (34), we strategy profile \((A, \mathcal{R}, \alpha)\) or equivalently \((n_a, \hat{n} - n_a, \alpha)\) is a pure Nash equilibrium if

\[ U_i(\text{"}'a', \alpha) = U_i(\text{"}'r', \alpha) \] (35)

That is

\[ \alpha P_{\text{suc}}(\text{"}'a', n_a) - \beta(1 - P_{\text{suc}}(\text{"}'a', n_a)) - \sigma_h = - \alpha P_{\text{suc}}(\text{"}'a', n_a) - \gamma \] (36)

The source utility writes

\[ U_i(\text{"}'a', \alpha) = (\hat{n} - n_a P_{\text{suc}}(\text{"}'a', n_a) + 1) \alpha - \alpha_{\text{max}} (\hat{n} - 2 n_a P_{\text{suc}}(\text{"}'a', n_a)) - \hat{n} \sigma_t \]

which is a linear function; hence, it can reach its maximum:

1. For \( \alpha = 0 \), if its slope is negative, that is

\[
\hat{n} - n_a (P_{\text{suc}}(\text{"}'a', n_a) + 1) \leq 0
\]

\[ \Rightarrow n_a \geq \frac{\hat{n}}{(P_{\text{suc}}(\text{"}'a', n_a) + 1)} \]

Substituting \( \alpha = 0 \) into equation (36) yields

\[ P_{\text{suc}}(\text{"}'a', n_a) - \sigma_h = - \gamma \]

\[ \Rightarrow P_{\text{suc}}(\text{"}'a', n_a) = \frac{\lambda (\beta - \gamma) + \sigma (1 - Q_h)}{\lambda \beta} \]

which completes the proof of the first condition.

2. For \( \alpha = \alpha_{\text{max}} \), if its slope is positive, that is

\[
\hat{n} - n_a (P_{\text{suc}}(\text{"}'a', n_a) + 1) \geq 0
\]

\[ \Rightarrow n_a \leq \frac{\hat{n}}{(P_{\text{suc}}(\text{"}'a', n_a) + 1)} \]

Substituting \( \alpha = \alpha_{\text{max}} \) into equation (36) yields
\[
\alpha_{\max}^2 \text{succ}(\text{"a"}, n_a) - \beta(1 - P_{\text{succ}}^{\text{suc}}(\text{"a"}, n_a)) - \sigma_h \\
= - \alpha_{\max}^2 \text{succ}(\text{"a"}, n_a) - \gamma
\]

Thus
\[
P_{\text{succ}}^{\text{suc}}(\text{"a"}, n_a) = \frac{\lambda(\beta - \gamma) + \sigma(1 - Q_h)}{\lambda(2\alpha_{\max} + \beta)}
\]

which completes the proof of the second condition.

**Mixed Nash equilibrium**

Now, as mixed strategies are allowed, the source/relay problems are as follows:

**Source problem.** The source node seeks to set the best reward value \(\alpha^{\text{BR}}(p)\), while relay nodes accept to cache the file with some probability \(p\).

**Relay problem.** Given a reward \(\alpha \in [0, \alpha_{\max}]\) offered by the source node, each relay seeks to compute the cooperating probability \(P^{\text{BR}}(\alpha)\) that maximizes its own objective function.

When a mixed Nash equilibrium is achieved, each relay node becomes indifferent about which strategy to choose. Namely
\[
U_i(p, \alpha^*) = U_i(1 - p, \alpha^*)
\]

Then
\[
\alpha^* P_{\text{succ}}(p^*, n) - \beta(1 - P_{\text{succ}}^{\text{suc}}(p^*, n)) - \sigma_h = - \alpha^* P_{\text{succ}}(p^*, n) - \gamma
\]

where
\[
P_{\text{succ}}^{\text{suc}}(p, n) = \frac{Z}{n} \sum_{j=1}^{n} \left\{ \frac{(Z-1)^{n-1} - (1-Z)^{n-j}}{n-1} \right\}
\]

with \(Z = p(1 - Q_h)\). Next, we solve equation (38) and obtain
\[
\alpha^* = \frac{\lambda \beta(n - 1 + (1 - Q_h)p^*) - \sigma(1 - Q_h)}{2\lambda(1 - (1 - Q_h)p^*)}
\]

Now, the expected utility of the source node becomes
\[
U_i(p, \alpha) = \sum_{n_a=0}^{n} \left( \frac{n}{n_a} \right) p^{n_a}(1 - p)^{n-n_a} U_i(\text{"a"}, \alpha)
\]

\[
= \sum_{n_a=0}^{n} \left( \frac{n}{n_a} \right) p^{n_a}(1 - p)^{n-n_a}
\]

\[
((n - n_a(p_{\text{succ}}^{\text{suc}}(\text{"a"}, n_a) + 1))\alpha - \alpha_{\max}(n - 2n_aP_{\text{succ}}^{\text{suc}}(\text{"a"}, n_a)) - \sigma_h = (n - 1) + \sum_{n_a=0}^{n} \left( \frac{n}{n_a} \right) \left( \frac{Q_h}{p^*} \right)^{n_a}(1 - p)^{n-n_a}
\]

\[
- \alpha_{\max}(n - 2 + 2n_a)\left( \frac{Q_h}{p^*} \right)^{n_a}(1 - p)^{n-n_a} - n\sigma_h = (n - 1 + (1 - (1 - Q_h)p^*)^{n} - np^*) = 0
\]

At Nash equilibrium, the first-order derivative of the source utility vanishes. Namely
\[
\frac{\partial U_i(p^*, \alpha)}{\partial \alpha} = 0
\]

\[
n - 1 + (1 - (1 - Q_h)p^*)^{n} - np^* = 0
\]

Notice that the left side of previous equation (42) is continuous, strictly decreasing monotone function on \([0, 1]\) with \(\frac{\partial U_i(0, \alpha^*)}{\partial \alpha} = n >0\) and \(\frac{\partial U_i(1, \alpha^*)}{\partial \alpha} = Q_h^2 - 1 < 0\); then, \(\frac{\partial U_i(p^*, \alpha)}{\partial \alpha} = 0\) has an unique solution \(p^*\).

**Summary.** The caching game has a unique mixed Nash equilibrium fully characterized by
\[
\left\{ \alpha^* = \frac{\lambda \beta(n - 1 + (1 - Q_h)p^*) - \sigma(1 - Q_h)}{2\lambda(1 - (1 - Q_h)p^*)} \right\}
\]

\(p^*\) is the unique solution of equation (42)

**PoA**

The caching game has a unique Nash equilibrium, and thus, the PoA is written as
\[
\text{PoA} = \frac{W(p^*, \alpha^*)}{W(p^{\text{BR}}, \alpha^*)}
\]

where \(W(p, \alpha)\) is the sum of the expected utility functions (social welfare) which are defined by
$$W(p, \alpha) = U_i(p, \alpha) + \sum_{i=1}^{n} U_i(p, \alpha)$$

$$= (n-1 + (1 - (1 - Q_h)p)^n - np)\alpha$$

$$- \alpha_{\text{max}}(n - 2 + 2(1 - (1 - Q_h)p)^n)$$

$$- n\alpha_i + n(p((2\alpha + \beta)(1 - (1 - (1 - Q_h)p)^n))$$

$$- \beta - \sigma_h + \gamma$$

$$- \alpha(1 - (1 - Q_h)p)^n - \gamma)$$

The pair $$(p^o, \alpha^o)$$ denotes the maximizer of social welfare function.

**Learning algorithms and insights for real-world implementation**

We exhibit here some discussions and algorithms on the implementation of our incentive mechanism in real-world networks. The proposed reward mechanism can be efficiently and simply implemented through distributed learning algorithm. Obviously, due to the lack of the acknowledgement (ACK) about the state of the transmission (success or failure), which may incur into large delays, we propose distributed learning algorithms for the relays and the source to allow them to learn their individual best decisions only based on information available locally and independently over time, that is, they use the local observations to estimate their payoffs. But, once a successful delivery occurs, each node receives its real reward allowing to correct the learning trajectory.

**Learning scheme for the source node**

The source chooses a real value $$\alpha$$ at each instant $$k$$ depending on the current strategy probability distribution of relays, and it has a problem to provide the optimal value of $$\alpha$$ that maximizes its payoff. Then, it picks this value continually and independently from interval $$[0, \alpha_{\text{max}}]$$ based on its local observations and its estimate to its own payoff at each learning step. In order to solve the source optimization problem, we suggest a stochastic approximation scheme. For instance, we consider the Kiefer–Wolfowitz scheme, the reader is invited to check Sastry and Thathachar and Ezzahidi et al. for more details.

**Learning schemes for the relay node**

The relay’s problem consists of choosing their best response strategy (to face the source policy, but also to face the other relays policies) that allows them to maximize their payoffs over time. At any time slot, every relay node selects a pure strategy according to a probability distribution. In order to discover pure Nash equilibrium, we consider the well-known Linear Reward-Inaction (LRI) algorithm. The LRI scheme only requires a slight amount of local information. For instance, it only requires the previous value of the distribution probability, the actual realization of selected pure strategy, and the reward received at previous step. Since LRI can only discover pure equilibria, we suggest a modified version called Linear Reward $$\epsilon$$-Penalty $$(L_{R-\epsilon\text{P}})$$ algorithm to reach mixed Nash equilibrium.29

**Source node: learning continuous action**

In previous sections “Two-person caching game” and “The $$n$$-person caching game,” we showed that from a designer perspective, the source objective function $$U_i$$ has a unique maximizer at $$\alpha^*$$ in a realistic network, the source node is neither aware of its objective function nor of the structure of the Nash equilibrium. However, the source node receives periodically a realization of its actual reward function as numeric value. Latter realization is a local observation $$\hat{U}_i(\hat{\alpha})$$ that may help the source node to construct its own utility function and thereby learn its equilibrium strategy. The Kiefer–Wolfowitz stochastic approximation algorithm uses the instantaneous estimate of the gradient to adjust/update the learning scheme. Thus, whenever the learning trajectory goes in the right direction, the gradient keeps being positive. Otherwise, the gradient becomes negative and the dynamics bring the trajectory to the right direction. Convergence is guaranteed when the gradient vanishes. The update rule is given by the following

$$\hat{\alpha}_{k+1} = \hat{\alpha}_k + \mu_k \nabla \hat{U}_i(\hat{\alpha}_k)$$

(45)

where the estimated gradient is given by

$$\nabla \hat{U}_i(\hat{\alpha}_k) = \frac{1}{c_k} (\hat{U}_i(\hat{\alpha}_k + c_k) - \hat{U}_i(\hat{\alpha}_k - c_k))$$

If the step sizes $$\mu_k$$ and $$c_k$$ satisfy the following

$$\begin{cases} \sum_k \mu_k = \infty \\ \sum_k \mu_k c_k < \infty \\ \sum_k \left( \frac{\mu_k}{c_k} \right)^2 < \infty \\ \mu_k < 0 \gamma \epsilon^2 < \gamma - \beta (1 - P_{\text{succ}}(\text{“a”}, \hat{r})) \end{cases}$$

(46)

where $$0 < \rho \epsilon < \gamma$$; hence, the algorithm converges to local maximum of $$\hat{U}_i$$ under some conditions on the function $$U_i$$ and the variance of the observations $$\hat{U}(\hat{\alpha})$$.28 The usual choices for step sizes are $$c_k = c_1/k^2$$, $$\mu_k = \mu_1/k$$ with $$0 < \gamma < 1/2$$, $$\mu_1 > 0$$, $$c_1 > 0$$. 

For instance, the LRI algorithm converges to local maximum of $$\hat{U}_i$$ under some conditions on the function $$U_i$$ and the variance of the observations $$\hat{U}(\hat{\alpha})$$.28
Relay node: learning pure Nash equilibrium

We use the LRI algorithm for the relays to learn their respective best response strategy (pure strategy only). Every relay updates a probability distribution independently over time with a view to converge to the best strategy progressively. Initially, the relay chooses strategy based on the initial probability distribution; after each time instant, the algorithm increases the probability to pick strategies with high observed utility and decreases the probability to choose the other strategies. Let $p_k = (p_k, 1 - p_k)$ be the probability distribution at instant $k$, where $p_k$ denotes the probability to accept the source deal at time instant $k$. The update rule is written as

$$p_{k+1} = p_k + \eta_k U_i(e_k, \hat{a})(e_k - p_k)$$

(47)

where $e_k$ is the indicator function which equals $1$ when the action “a” (accept the source deal) is selected at time instant $k$, otherwise it equals $0$. And $0 < \eta_k < 1$ is the learning speed at time instant $k$. One can easily check that these dynamics converge to pure Nash equilibria when it exists. However, unfortunately, it may converge to pure profiles that are not equilibria. This is why the initialization phase is crucial for such a learning scheme.

Relay node: learning mixed Nash equilibrium

Ultimately, the derived mixed Nash equilibrium exhibits some nice features. In particular, it has better energy efficiency and better fairness properties since each relay node may participate in the caching game. To guarantee convergence to such an equilibrium, we propose the LRI algorithm to drive the relay nodes to their respective equilibrium acceptance probabilities $p^* = (p_1^*, p_2^*, \ldots, p_m^*)$. Practically, every relay node updates its respective distribution probability, independently of its opponents, over time according to following rule

$$p_{k+1} = p_k - \eta_1 U_i(e_k, \hat{a})(e_k - p_k) + \eta_2 (1 - U_i(e_k, \hat{a})) \left( \frac{e_k}{m - e_k} - p_k \right)$$

(48)

where $\eta_2 = \epsilon \cdot \eta_1$, $\epsilon < 1$ is a small real number, and $m = 2$ is the number of relay strategy (either “a” or “r”). At each iteration, the relay randomly chooses a strategy according to its own distribution probability and updates the strategy probability vector at instant $k$ until convergence, that is, until achieving $p^* \in \arg \max_p U_i(\alpha^*, p)$.

Numerical investigations

In this section, we present some numerical results and simulation runs to illustrate the behavior of our algorithms. Unless contrary indication, we use the following setting: $a_{\text{max}} = 3$, $\beta = 0.04$, $\sigma = 0.25$, $\sigma_l = 0.02$, and $\gamma = 0.05$. Next, we plot the performance metrics of the caching game Nash equilibria while varying the horizon $h$ (file lifetime) and the mobility parameter $\lambda$ quantifying the contact rate. The considered metrics are the caching acceptance probability $p^*$, the equilibrium reward value $\alpha^*$, the utility functions, and the PoA.

Figure 2(a) and (b) depicts the probability of acceptance $p^*$ and the file value $\alpha^*$ at Nash equilibria as function of the file lifetime $h$. We note that the file value (reward given to the relay node) increases as the horizon increases. The relay node meanwhile tends to decrease its willingness to cache the file as the value of $\alpha$ increases. This behavior can be explained as follows. On one hand, as the file lifetime increases, the relay nodes have less incentive to accept the caching offer since it is consuming too much storing energy. On the other hand, the source node seeks to convince the relay to cache the file by offering better deals with higher reward value. Moreover, we notice that the source has
incentive to offer less reward as the contact rate increases, which is quite intuitive since the caching transaction would not consume too much energy, whereas the relay nodes seem to behave a bit in a counter-intuitive way. Indeed, the relays have incentive to accept caching the file as the probability to contact the destination is low. This can be easily captured by the increasing reward offered by the source node and the low energy investment as the contact probability decreases.

We plot in Figure 3(a) and (b) the equilibrium payoff functions of the source and the relay (respectively) over file lifetime $h$. For low values of $h$, the source utility increases until a given threshold $h_{th} = h(\lambda)$ and converges to the maximum payoff value $U_s^* = \lim_{h \to \infty} U_s(\alpha^*, p^*) = -9$. We remark that the threshold horizon $h_{th}$ increases as the inter-contact rate $\lambda$ decreases. As for a relay node, we notice that its utility is strictly decreasing as $h$ goes to infinity. A special feature is that for both nodes, it is more interesting to consider long enough file lifetime with some reasonable contact probability.

Figure 4(a) and (b) shows the impact of the network density (the number of relay nodes) on the equilibrium probability of acceptance and file value. We notice that $p^*$ and $\alpha^*$ both increase as the number of relay nodes increase. When the network becomes dense/ultra-dense, the competition among relay nodes is getting fierce. So, the probability to be the first to deliver the file content is getting smaller and smaller. This may push relay nodes to often decline the source deal, so the source node has interest in increasing the reward it offers. This improves again the acceptance probability. We find out again that the relays are pretty interested in the file with low lifetime.

Next, we depict in Figure 5(a) and (b) the impact of the network density on the reward value and the probability of acceptance for different contact rates $\lambda$. Now, the source node is offering higher reward for low value of $\lambda$, which is quite intuitive. This is indeed an incentive action to ensure an acceptable delivery probability though the network topology is not changing enough to create contact opportunities. Moreover, the value of reward increases until a given threshold $n_{th} = n(\lambda)$ and becomes $\alpha_{max} = 3$. It is definitely a surprising feature that this saturation threshold increases as the $\lambda$ increases. This means that the source node still can manage to pay less as the network size is high, taking benefit from relay nodes competition.
Now, we focus on the loss of efficiency experienced at Nash equilibrium. We depict in Figure 6 the PoA while changing the file lifetime $h$ for different values of contact rate $\lambda$. We note that the PoA is very close to 1 for average and high values of $\lambda$, which means that the equilibrium performs as well as the social optimum. However, we remark that the PoA is bounded for low values of $\lambda$, and the loss of efficiency is not very significant which means that our scheme drives the system to even a near-optimal point. One notices also that increasing the file lifetime $h$ may increase the equilibrium efficiency. Hence, a good efficiency could be met using some cross-layer optimization including the protocol stacks and the network topology. For instance, one can adjust the file lifetime (application layer) and the mobility parameters (physical layer and network topology).

The remaining is devoted to illustrate the behavior of our learning algorithms discussed above in section “Learning algorithms and insights for real-world implementation.” We consider $h = 100$, $\lambda = 0.5$, and would like to mention that at least 10 runs have been averaged to produce the learning trajectories. Figure 7(a) and (b) shows a setup where the pure equilibrium $(A_0, R_0, 0)$ is discovered. Next, Figure 8(a) and (b) indicates the convergence to the pure equilibrium $(A_0, R_k, \alpha_{\text{max}})$. We note that the probability of acceptance of five relays converges to the same probability either 0 or 1 according to the considered setting. A convergence illustration of the
mixed equilibrium \((p^*, \alpha^*)\) is depicted in Figure 9(a) and (b). The three algorithms exhibit impressive properties in terms of convergence speed and convergence accuracy. However, a theoretic convergence analysis is still worth consideration and is crucial to properly address all convergence-related issues. This would in particular optimize implementation efforts and make such an incentive mechanism more efficient.

**Conclusion**

In this work, we propose a reward-based incentive mechanism for file caching for DTN environment under constraints of energy consumption, file lifetime, and rate of contact. The source–relay interaction is captured using a non-cooperative game. Next, we give some necessary conditions for the existence of pure Nash equilibria. Then, we explicitly derive the unique mixed equilibrium of this caching game which seems to have some very nice features. For instance, it exhibits better fairness properties and improved energy efficiency in comparison with pure equilibria. Furthermore, we propose three fully distributed algorithms to discover the pure equilibrium and the mixed equilibrium as well. The first scheme allows the source node to learn its equilibrium strategy (continuous action). The second scheme is used by relay nodes to learn their respective pure strategies (discrete strategy). Whereas the third scheme is designed to explore the relay nodes’ mixed strategies at equilibrium.

As a part of our future works, we aim to implement such a mechanism in a real-world network using DTN framework. We are also interested in building a more energy-efficient mechanism by allowing relay nodes to switch to sleep mode to save energy while allowing a delivery probability higher than some given threshold.

**Acknowledgements**

The authors thank the anonymous referees and the editor for their constructive comments and suggestions to improve the presentation of this article.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.
Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work has been conducted within the framework of the Mobicty Project funded by the Moroccan Ministry of Higher Education and Scientific Research and the National Centre for Scientific and Technical Research.

References
1. Fall K. A delay-tolerant network architecture for challenged internets. In: Proceedings of the 2003 conference on applications, technologies, architectures, and protocols for computer communications. Karlsruhe, 25–29 August 2003, pp.27–34. New York: ACM.
2. Krishnan R, Basu P, Mikkelsen JM, et al. The spindle disruption-tolerant networking system. In: Proceedings of the IEEE military communications conference (MILCOM 2007), Orlando, FL, 29–31 October 2007, pp.1–7. New York: IEEE.
3. Burleigh S, Hooke A, Torgerson L, et al. Delay-tolerant networking: an approach to interplanetary internet. IEEE Commun Mag 2003; 41(6): 128–136.
4. Partan J, Kurose J and Levine BN. A survey of common issues in underwater networks. ACM SIGMOBILE Mob Comput Commun Rev 2007; 11(4): 23–33.
5. Bastani S, Landfeldt B and Libman L. On the reliability of safety message broadcast in vehicular urban ad hoc networks. In: Proceedings of the 14th ACM international conference on modeling, analysis and simulation of wireless and mobile systems, Miami, FL, 31 October–4 November, pp.307–316. New York: ACM.
6. Jiang X and Bai XY. A survey on incentive mechanism of delay tolerant networks. In: Proceedings of the 10th international conference on computer systems and network technologies (ICCSNT). Chengdu, China, 17–19 December 2013, pp.191–197. New York: IEEE.
7. Hulke SP and Attar V. Game theory based incentive mechanisms in delay tolerant network. Int J Comput Appl T 2015; 115(1): 22–25.
8. Chahin W, Sidi HB, El-Azouzi R, et al. Incentive mechanisms based on minority games in heterogeneous delay tolerant networks. In: Proceedings of the 25th international teletraffic congress (ITC), Shanghai, China, 10–12 September 2013, pp.1–9. New York: IEEE.
9. El-Azouzi R, De Pellegrini F, Sidi HB, et al. Evolutionary forwarding games in delay tolerant networks: equilibria, mechanism design and stochastic approximation. Comput Netw 2013; 57(4): 1003–1018.
10. Brun O, El-Azouzi R, Prabhu B, et al. Modeling rewards and incentive mechanisms for delay tolerant networks. In: Proceedings of the 12th international symposium on modeling and optimization in mobile, ad hoc, and wireless networks (WiOpt), Hammamet, Tunisia, 12–16 May 2014, pp.233–240. New York: IEEE.
11. Lu Y, Wang W, Chen L, et al. Opportunistic forwarding in energy harvesting mobile delay tolerant networks. In: Proceedings of the IEEE international conference on communications (ICC), Sydney, NSW, Australia, 10–14 June 2014, pp.526–531. New York: IEEE.
12. Niyato D, Wang P, Tan HP, et al. Cooperation in delay-tolerant networks with wireless energy transfer: performance analysis and optimization. IEEE T Veh Technol 2015; 64(8): 3740–3754.
13. Neglia G and Zhang X. Optimal delay-power tradeoff in sparse delay tolerant networks: a preliminary study. In: Proceedings of the 2006 SIGCOMM workshop on challenged networks, Pisa, 11–15 September 2006, pp.237–244. New York: ACM.
14. Krifa A, Barakat C and Spyropoulos T. Message drop and scheduling in DTNs: theory and practice. IEEE T Mobile Comput 2012; 11(9): 1470–1483.
15. Wang T, Hui P, Kulkarni S, et al. Cooperative caching based on file popularity ranking in delay tolerant networks (arXiv preprint arXiv:14097047), 2014, https://arxiv.org/abs/1409.7047.
16. Koulali S, Sabir E, Taleb T, et al. A green strategic activity scheduling for UAV networks: a sub-modular game perspective. IEEE Commun Mag 2016; 54(5): 58–64.
17. Le T, Lu Y and Gerla M. Social caching and content retrieval in disruption tolerant networks (DTNs). In: Proceedings of the international conference on computing, networking and communications (ICNC), Anaheim, CA, 16–19 February 2015, pp.905–910. New York: IEEE.
18. Yin L and Cao G. Supporting cooperative caching in ad hoc networks. IEEE T Mobile Comput 2006; 5(1): 77–89.
19. Harra T. Effective replica allocation in ad hoc networks for improving data accessibility. In: Proceedings of the twentieth annual joint conference of the IEEE computer and communications societies (INFOCOM 2001) (vol. 3), Anchorage, AK, 22–26 April 2001, pp.1568–1576. New York: IEEE.
20. Fiore M, Mininni F, Casetti C, et al. To cache or not to cache? In: Proceedings of the INFOCOM 2009, Rio de Janeiro, Brazil, 19–25 April 2009, pp.235–243. New York: IEEE.
21. Al Hanbali A, Nain P and Altman E. Performance of ad hoc networks with two-hop relay routing and limited packet lifetime (extended version). Perform Eval 2008; 65(6): 463–483.
22. Li Y, Su G, Wu DO, et al. The impact of node selfishness on multicasting in delay tolerant networks. IEEE T Veh Technol 2011; 60(5): 2224–2238.
23. Ezzahidi SA, Sabir E and Bouyakhf EH. A reward-based incentive mechanism for file caching in delay tolerant networks. In: Proceedings of the IEEE/IFIP network operations and management symposium, Istanbul, Turkey, 25–29 April 2016, pp.863–867. New York: IEEE.
24. Ezzahidi SA, Sabir E, Kamili ME, et al. A non-cooperative file caching for delay tolerant networks: a reward-based incentive mechanism. In: Proceedings of the 2016 IEEE wireless communications and networking conference, Doha, Qatar, 3–6 April 2016, pp.1–6. New York: IEEE.
25. Gao W, Li Q, Zhao B, et al. Multicasting in delay tolerant networks: a social network perspective. In: Proceedings of the tenth ACM international symposium on mobile communications (IMC), Sydney, NSW, Australia, 10–14 June 2014, pp.526–531. New York: IEEE.
ad hoc networking and computing, New Orleans, LA, 18–21 May 2009, pp.299–308. New York: ACM.
26. Altman E. Competition and cooperation between nodes in delay tolerant networks with two hop routing. In: Núñez-Queija R and Resing J (eds) Network control and optimization. Berlin: Springer, 2009, pp.264–278.
27. Koutsoupias E and Papadimitriou C. Worst-case equilibria. In: Meinel C and Tison S (eds) STACS 99. Berlin: Springer, pp.404–413.
28. Kiefer J and Wolfowitz J. Stochastic estimation of the maximum of a regression function. Ann Math Stat 1952; 23(3): 462–466.
29. Thathachar MA and Sastry PS. Networks of learning automata: techniques for online stochastic optimization. Berlin: Springer science + Business Media, 2011.