
Enas Odat\(^a\), Doha Hamza\(^a\), Basem Shihada\(^a\), Jeff S. Shamma\(^a\)

\(^a\)Department of Electrical Engineering, King Abdullah University of Science and Technology, Thuwal, 23955-6900, Saudi Arabia

Abstract

The Internet of Things (IoT) paradigm envisions billions of interconnected things. This high density of things generates a huge amount of data, which results in communication delays and increases the amount of contention. Thus, it is necessary to have efficient Medium Access Control (MAC) protocols to coordinate channel access. In this paper, we consider a network throughput maximization problem in which a set of selfish nodes compete for transmission opportunities in the IoT scenario. To enhance cooperation among the nodes and to reduce the collision rate, we formulate a memory-one channel access game in which the nodes maximize their payoffs by optimizing their channel access probabilities, conditioned on their previous transmission state. A distributed learning algorithm is used by the nodes to solve the problem efficiently in order to overcome any coordination overhead. We investigate the impact of the network topology on the solution to the problem. Our simulation results show that the throughput achieved by the memory-one game outperforms the throughput achieved by other methods including IEEE 802.11 DCF protocol.

Keywords: IoT, Random access games, Perfect information, Gradient play, Imperfect information.
1. Introduction and Related Work

Advances in technology and the emergence of low complexity intelligent devices resulted in the evolution of the Internet-of-Things (IoT). In most IoT application scenarios, billions of things are interconnected together using standard communication protocols to provide services for different applications such as in the healthcare industry, smart cities, transportation, and food supply chain [1]. Despite their potential of connecting things anywhere, anytime, and anyplace, IoT devices present many challenges due to the heterogeneity, density, and the power constraints of things, and the intermittent nature of the network where things might connect and disconnect at any time. All of these factors increase the communication delay and the generated data, and it is thereby necessary to develop resource management solutions for IoT applications.

One of the important resources is the wireless medium, which is a shared resource; thus, it is necessary for the nodes to have methods that schedule channel access. Current access methods can be classified into two categories: contention-free and contention-based. The main difference between these methods is the way by which they coordinate the nodes; for example, in contention-free methods, there is a centralized entity that schedules the nodes for access, whereas in the distributed contention-based methods, the nodes compete for channel access. This lack of explicit coordination in the contention-based methods reduces the coordination overhead compared to the contention-free methods. However, it increases the complexity of designing efficient channel access protocols that maximize the network-wide performance, such as the throughput and that scales well with the increase in the number of nodes.

An example of contention-based access methods is the Carrier Sense Multiple Access (CSMA) protocol, where a node senses the channel before its transmission and waits for a random amount of time if the channel is sensed to be busy. CSMA algorithms are the main mechanism of the IEEE 802.11 Distributed Coordination Function (DCF) medium access control protocol. In this protocol, the nodes use the exponential backoff algorithm (EB), where each of them doubles its contention window size (up to maximum size) after a collision and resets it to the minimal size after a successful transmission [2]. There is a fairness issue in this protocol since the nodes that back
off exponentially will most likely have to wait for a longer time duration, while other nodes are accessing the channel during that time. A solution to this problem is to use the same contention window size for all nodes in each time step instead of using a variable contention window size as in the DCF protocol [3], [4], and [5].

The aforementioned fairness issue and the selfish behavior of the nodes motivate our design of a game-theoretical solution to model the contention-based medium access protocol. In such models, the nodes work to maximize their payoffs and update their transmission probabilities in response to the contention level in the network. In [6], a game theory approach is proposed to find the optimal contention window size. In that work, each node maximizes its own payoff and updates its transmission probability in response to the contention level in the network using gradient play. It is shown that the network reaches a symmetric Nash equilibrium which is a fair solution for all nodes. Compared to this approach, our goal is to induce cooperation among the nodes by introducing memory into the channel access game, where each node adjusts its transmission probability based on the outcome of the last transmissions of nodes in the network.

To illustrate the advantage of our proposed method, we use the following example. Consider a simple network of two nodes where each one of them transmits with probability $p = 0$ if it successfully transmitted in the previous time slot, and with $p = 1$ if the other node successfully transmitted in the previous time slot. As a result, the collision among the transmission of both nodes is ideally zero, since both nodes alternate their transmissions by using such a memory-based strategy. In contrast to the work in [6], where a node accesses the channel at each time slot using the same transmission probability, $p \in [0, 1]$, regardless of the channel state in the previous time slot, this example suggests that introducing memory-one strategies to the channel access game induces an equilibrium point where nodes cooperate to avoid collisions and achieve higher throughput on average.

The rest of the paper is organized as follows. We first describe the system model and define the throughput maximization problem (Section 2). We then formulate the memory-one channel access game in the case of perfect information scenario where we assume that each node knows the transmission probabilities or the strategies of other
nodes in the network. The perfect information scenario becomes unrealistic when considering a large number of nodes. Thus, we further study the channel access game in the imperfect information scenario, where each node knows its own transmission probability vector only (Section 3). In this case, each node observes the channel state resulting from the interaction of all nodes in the network and uses these observations to estimate the transmission probabilities of other nodes. We use the normalized throughput to compare the performance of different methods and extend the Bianchi model to find the parameters of the IEEE 802.11 DCF protocol for a network with general topology (Section 4). We numerically evaluate the performance of the memory-one channel access game and compare it with other methods (Section 5). Finally, we conclude the paper (Section 6).

2. System Model and Problem Formulation

Consider a network of selfish nodes that are competing to access the channel for a transmission. The topology of this network is modeled as an undirected interference graph \( G = \{N, \varepsilon\} \), where each vertex \( i \in N \) is a link of sender and receiver pair, and each edge \( \varepsilon_{ij} \in \varepsilon \) indicates that the transmission of link \( i \) interferes with the transmission of link \( j \) [7]. In the case of a fully connected interference graph, all links in the network are in the interference range of each other. Other examples of interference graphs include ring and line graphs.

Throughout our analysis, we assume that the nodes are backlogged, which means that they always have packets for transmission. We assume a time-slotted system model, where at each time slot \( t \), node \( i \) attempts to transmit a packet with a transmission probability that depends on the channel state which we define below. The joint outcome (actions) of the transmission attempts of all the nodes in the network at time \( t \) is denoted by \( S(t) = \prod_{i \in N} a_i \), where \( a_i(t) \in \{0, 1\} \). Specifically, 0 means that node \( i \) did not transmit, whereas 1 means that node \( i \) transmitted at time \( t \). Each tuple \( S(t) \) represents the actual state of all nodes in the network at time \( t \). We further define the
channel state at time $t$ $s_i(t)$ from the point of view of node $i$ as follows:

\[
s_i(t) = \begin{cases} 
11, & \text{if } a_i(t) = 1, a_j(t) = 1, \text{for any } j \in \mathcal{N}(i) \\
10, & \text{if } a_i(t) = 1, a_j(t) = 0, \text{for all } j \in \mathcal{N}(i) \\
01, & \text{if } a_i(t) = 0, a_j(t) = 1, \text{for any } j \in \mathcal{N}(i) \\
00, & \text{if } a_i(t) = 0, a_j(t) = 0, \text{for all } j \in \mathcal{N}(i)
\end{cases}
\]  

(1)

where $\mathcal{N}(i)$ is the set of node $i$ neighbors, state $s_i(t) = \{11\}$ corresponds to a collision in the neighborhood of node $i$, $s_i(t) = \{10\}$ denotes a successful transmission for node $i$ at time $t$, state $s_i(t) = \{01\}$ indicates that node $i$ did not transmit but some other neighboring node transmitted at time $t$, and finally $s_i(t) = \{00\}$ indicates the backoff of node $i$ and all nodes in its neighborhood.

Let us define $\mathbf{p}_i = (p_{11}^{i}, p_{10}^{i}, p_{01}^{i}, p_{00}^{i})$ where $p_{s(t)}^{i} \in [p_{\min}, p_{\max}], 0 < p_{\min} < p_{\max} < 1$, is node $i$’s transmission probability at time $(t + 1)$ conditioned on the channel state $s_i(t)$ at time $t$. These probabilities imply a Markov chain whose stationary distribution defines the probabilities of the channel states. In the following subsection, we present the model of the channel-state transition matrix required to compute the stationary distribution of the channel states.

**Example 1.** Consider a set of 3 nodes placed in a network of line topology.

```
1 ——— 2 ——— 3
```

In this network:

\[
\{S(t)\} = \{000, 001, 010, 011, 100, 101, 110, 111\},
\]

\[\mathcal{N}(2) = \{1, 3\}, \mathcal{N}(1) = \{2\}, \mathcal{N}(3) = \{2\}\]

\[\{000, 001\} \in s_1(t) = 00, \{010, 011\} \in s_1(t) = 01, \{100, 101\} \in s_1(t) = 10, \]

\[\{110, 111\} \in s_1(t) = 11,\]

\[\{000\} \in s_2(t) = 00, \{001, 100, 101\} \in s_2(t) = 01, \{010\} \in s_2(t) = 10, \]

\[\{110, 111, 011\} \in s_2(t) = 11,\]

\[\{000, 100\} \in s_3(t) = 00, \{010, 110\} \in s_3(t) = 01, \{001\} \in s_3(t) = 10, \]

\[\{011, 111\} \in s_3(t) = 11.\]
2.1. Channel-State Transition Matrix for Two Nodes

We assume that the states of the channel evolve over time according to a Markov chain where the state at time \((t + 1)\) depends on the state at time \((t)\) and the joint action of the nodes. This evolution is modeled by a channel-state transition matrix. Consider a network consisting of two nodes \(i\) and \(j\). The entries of the channel-state transition matrix of node \(i\) can be computed as

\[
\begin{align*}
\mathbb{P}(s_i(t+1) = 11|s_i(t)) &= p^{s_i(t)}_i p^{s_j(t)}_j \\
\mathbb{P}(s_i(t+1) = 10|s_i(t)) &= p^{s_i(t)}_i (1 - p^{s_j(t)}_j) \\
\mathbb{P}(s_i(t+1) = 01|s_i(t)) &= (1 - p^{s_i(t)}_i) p^{s_j(t)}_j \\
\mathbb{P}(s_i(t+1) = 00|s_i(t)) &= (1 - p^{s_i(t)}_i) (1 - p^{s_j(t)}_j).
\end{align*}
\]

Using this set of equations, the transition probability matrix \(\pi_i\), i.e. the transition probability matrix from the point of view of node \(i\), is given by

\[
\begin{pmatrix}
11 & 10 & 01 & 00 \\
11 & (p^{11}_i p^{11}_j) & p^{11}_i (1 - p^{11}_j) & (1 - p^{11}_i) p^{11}_j & (1 - p^{11}_i) (1 - p^{11}_j) \\
10 & p^{10}_i p^{01}_j & p^{10}_i (1 - p^{01}_j) & (1 - p^{10}_i) p^{10}_j & (1 - p^{10}_i) (1 - p^{01}_j) \\
01 & p^{01}_i p^{10}_j & p^{01}_i (1 - p^{10}_j) & (1 - p^{01}_i) p^{10}_j & (1 - p^{01}_i) (1 - p^{10}_j) \\
00 & p^{00}_i p^{00}_j & p^{00}_i (1 - p^{00}_j) & (1 - p^{00}_i) p^{00}_j & (1 - p^{00}_i) (1 - p^{00}_j)
\end{pmatrix}
\]

2.2. Channel-State Transition Matrix for Multiple Nodes

We extend the model of two nodes to the case of \(N\) nodes, where the entries of the channel-state transition matrix, \(\pi_i\), are given by

\[
p(S(t+1)|S(t)) = \prod_{i \in \mathcal{N}} [(1 - p^{S_i(t)}_i) \mathbb{I}(a_i(t+1) = 0) \\
+ (p^{S_i(t)}_i) \mathbb{I}(a_i(t+1) = 1)]
\]
where $S(t) \in \prod_{i \in \mathcal{N}} a_i(t), S_i(t) = p_i S(t), \forall S(t) \in s_i(t)$, and $1$ is the indicator function.

**Example 2.** Consider the network in Example 1. The following are the results of applying Equation 3:

$$p(S(t+1) = 110 | S(t) = 010) = p_1^{01} \cdot p_2^{10} \cdot (1 - p_3^{01})$$

$$p(S(t+1) = 010 | S(t) = 011) = (1 - p_1^{01}) \cdot p_2^{11} \cdot (1 - p_3^{11})$$

### 2.3. Stationary Distribution of the Channel State

The transmissions of the nodes at time $t$ are random variables that have a set of possible values, or the state space, $\{S(t)\}$ and $\{s_i(t)\}$. To define discrete time-homogeneous Markov chains over this state space, we start with the following assumption:

**Assumption 1.** The Markov chains defined by the transition probability matrices $\pi$ is irreducible and aperiodic.

Note that Assumption 1 guarantees that the corresponding Markov chain has a unique stationary distribution $\mu$, given by

$$\mu = 1^T \cdot (\pi + 11^T - I)^{-1} 1 \in \mathbb{R}^{2N}, I \in \mathbb{R}^{2N \times 2N}.$$  \hspace{1cm} (4)

**Example 3.** Consider the network in Example 1. $\mu$ is given by,

$$\mu = (\mu^{000}, \mu^{001}, \mu^{010}, \mu^{011}, \mu^{100}, \mu^{101}, \mu^{110}, \mu^{111}).$$

### 2.4. Aggregation of Channel-State Transition Matrix

The computation of the matrix $\pi_i \in \mathbb{R}^{2N \times 2N}$ requires a centralized controller that has perfect information about the actions of all nodes in the network, which becomes

---

1. Here and in Algorithm 1 we use the indicator function $1$ defined as follows. Suppose that $A \subset \omega$, and let $1_A : \omega \rightarrow \{0, 1\}$ be the indicator function of that set; i.e., $1_A(w) = 1$, if $w \in A$, and $1_A(w) = 0$, otherwise.
computationally prohibitive as the number of nodes in the network increases. Thus, we
reduce this matrix to $\pi^\text{red}_i \in \mathbb{R}^{4 \times 4}$ obtained by aggregating the states from $\{S(t)\}$ that
belong to the same type of state $s_i(t)$. As a result of this aggregation, the computation
of $\pi^\text{red}_i$ scales with the increasing number of nodes, and such a simplification enhances
the efficiency of the distributed implementation of our solution. To proceed, we define
the membership matrix $\phi_i \in \mathbb{R}^{2N \times 4}$ as follows:

$$
\phi_{kj} = \begin{cases} 
0, & \text{if } S_k(t) \notin s_i(t) \\
1, & \text{if } S_k(t) \in s_i(t) 
\end{cases}
$$

We use this matrix to write $\pi^\text{red}_i$ as follows:

$$
\pi^\text{red}_i = (\phi_i^T \text{diag}(\mu) \phi_i)^{-1} \phi_i^T \text{diag}(\mu) \pi \phi_i.
$$

**Definition 2.1.** Given a connected network $G = \{\mathcal{N}, \mathcal{E}\}$, we define the transition prob-
ability matrix $\pi^\text{red}_i$ for the Markov chain on the state space $s_i(t) \in \{11, 10, 01, 00\}$. The
entries of the $\pi^\text{red}_i$ are given by equation (6).

Before we proceed, we make the following assumption.

**Assumption 2.** The aggregated process $\{s_i(t)\}$ is an irreducible and aperiodic Markov
chain.

Assumption 2 guarantees that the corresponding Markov chain has a unique stationary
distribution $\mu^\text{red}_i$, given by

$$
\mu^\text{red}_i = 1^T (\pi^\text{red}_i + \mathbf{1}^T - \mathbf{1})^{-1} \mathbf{1} \in \mathbb{R}^4, \mathbf{1} \in \mathbb{R}^{4 \times 4},
$$

where the stationary distribution $\mu^\text{red}_i = (\mu^{11}_i, \mu^{10}_i, \mu^{01}_i, \mu^{00}_i)$ represents the probabilities
of different channel states $s_i(t)$ as follows: $\mu^{11}_i$ is the probability of collision; $\mu^{10}_i$ is the
probability of the successful transmission of node $i$ and backoff of other nodes in its
neighborhood; $\mu^{01}_i$ is the probability of backoff of node $i$ and the transmission by at
least one node in its neighborhood; $\mu^{00}_i$ is the probability of backoff by node $i$ and all
nodes in its neighborhood.

To illustrate Definition 2.1, we give the following examples.
Example 4. Consider a set of 3 nodes placed in a fully connected network.

We show some entries of the $\pi^\text{red}_i$ computed using equation (6).

\[
\begin{bmatrix}
11 & 10 & 01 & 00 \\
111 & 1 & 0 & 0 & 0 \\
110 & 1 & 0 & 0 & 0 \\
101 & 1 & 0 & 0 & 0 \\
100 & 0 & 1 & 0 & 0 \\
\end{bmatrix}
\]

$\phi_1 =
\begin{bmatrix}
11 & 10 & 01 & 00 \\
011 & 0 & 0 & 1 & 0 \\
010 & 0 & 0 & 1 & 0 \\
001 & 0 & 0 & 1 & 0 \\
000 & 0 & 0 & 0 & 1 \\
\end{bmatrix}$

\[
\pi^\text{red}_{11\rightarrow 11} = \frac{\mu_{111}}{\mu_{111} + \mu_{110} + \mu_{101}} [\pi_{111\rightarrow 111} + \pi_{111\rightarrow 110} + \pi_{111\rightarrow 101}]
+ \frac{\mu_{110}}{\mu_{111} + \mu_{110} + \mu_{101}} [\pi_{110\rightarrow 111} + \pi_{110\rightarrow 110} + \pi_{110\rightarrow 101}]
+ \frac{\mu_{101}}{\mu_{111} + \mu_{110} + \mu_{101}} [\pi_{101\rightarrow 111} + \pi_{101\rightarrow 110} + \pi_{101\rightarrow 101}]
\]

\[
\pi^\text{red}_{11\rightarrow 10} = \frac{\mu_{111} \pi_{111\rightarrow 100} + \mu_{110} \pi_{110\rightarrow 100} + \mu_{101} \pi_{101\rightarrow 100}}{\mu_{111} + \mu_{110} + \mu_{101}}
\]

\[
\pi^\text{red}_{10\rightarrow 11} = \pi_{100\rightarrow 111} + \pi_{100\rightarrow 110} + \pi_{100\rightarrow 101}
\]

Example 5. Consider a set of 3 nodes placed in a network of line topology.
We show some entries of the $\pi_{\text{red}}$ computed using equation (6).

\[
\begin{bmatrix}
11 & 10 & 01 & 00 \\
111 & 1 & 0 & 0 & 0 \\
110 & 1 & 0 & 0 & 0 \\
101 & 0 & 1 & 0 & 0 \\
100 & 0 & 1 & 0 & 0 \\
011 & 0 & 0 & 1 & 0 \\
010 & 0 & 0 & 1 & 0 \\
001 & 0 & 0 & 0 & 1 \\
000 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

$\phi_1 = \begin{bmatrix}
111 & 1 & 0 & 0 & 0 \\
011 & 0 & 0 & 1 & 0 \\
010 & 0 & 0 & 1 & 0 \\
001 & 0 & 0 & 0 & 1 \\
000 & 0 & 0 & 0 & 1 \\
\end{bmatrix}$

\[
\begin{align*}
\pi_{11\rightarrow11}^{\text{red}} &= \frac{\mu_{111}}{\mu_{111} + \mu_{110}} [\pi_{111\rightarrow111} + \pi_{111\rightarrow110}] \\
&\quad + \frac{\mu_{110}}{\mu_{111} + \mu_{110}} [\pi_{110\rightarrow111} + \pi_{110\rightarrow110}] \\
\pi_{11\rightarrow10}^{\text{red}} &= \frac{\mu_{111} \pi_{111\rightarrow100} + \mu_{110} \pi_{110\rightarrow100}}{\mu_{111} + \mu_{110}} \\
\pi_{10\rightarrow11}^{\text{red}} &= \frac{\mu_{100}}{\mu_{101} + \mu_{100}} [\pi_{100\rightarrow111} + \pi_{100\rightarrow110}] \\
&\quad + \frac{\mu_{101}}{\mu_{101} + \mu_{100}} [\pi_{101\rightarrow111} + \pi_{101\rightarrow110}] \\
\end{align*}
\]

2.5. Problem Formulation

Consider a network $\mathcal{G}$ with a set of nodes, where each node in the network maximizes its payoff which is defined as the utility of its successful transmission and is reduced by a cost that depends on the contention it causes to other nodes in the network. This is similar to the payoff function in [6]. Let us define $p = (p_i, p_{-i})$ as the strategy vectors of all nodes. The utility function, $U_i(\mu_i^{10}(p)) : p \rightarrow \mathbb{R}$, and the payoff function, $u_i(\mu_i^{10}(p), \mu_i^{11}(p)) : p \rightarrow \mathbb{R}$, of node $i$ are given by

\[
u_i(\mu_i^{10}(p), \mu_i^{11}(p)) = U_i(\mu_i^{10}(p)) - \mu_i^{11}(p) = \alpha \cdot \mu_i^{10}(p) + \beta \cdot \log(1 - \mu_i^{10}(p)) - \mu_i^{11}(p) \quad (8)
\]
where $\alpha > 0$, $\beta > 0$ are design parameters. In [6], $\alpha = (1 + e^{-\xi^*})$, $\beta = 2 \cdot e^{-\xi^*}$, where $\xi^*$ is a protocol dependent parameter. For example, its value is 0.1622 in the case of the IEEE 802.11b protocol. Based on this, we formulate the throughput maximization problem as follows:

$$\max_{p_i} u_i(\mu_i^{10}(p), \mu_i^{11}(p))$$

subject to

$$\mu_i^{\text{red}} = \mu_i \cdot \pi_i^{\text{red}},$$

$$0 \leq \mu_i^{s(t)} \leq 1,$$

$$\sum_{s(t)} \mu_i^{s(t)} = 1,$$

$$p_i^{\min} \leq p_i^{s(t)} \leq p_i^{\max}, \quad \forall i = 1, 2, 3, \ldots, N,$$

where $\mu_i^{s(t)}$ are the entries of $\mu_i^{\text{red}}$.

### 3. Channel Access Game

To solve the throughput maximization problem defined in Section 2.5, we apply a game theoretic method. Under the perfect information scenario, nodes solve problem in (9) using the strategy vectors $p$ that are available to them. Specifically, each node uses the strategy vectors $p$ to calculate the channel state transition matrix $\pi_i^{\text{red}}$ and find the stationary distribution $\mu_i^{\text{red}}$. Then, it uses $\pi_i^{\text{red}}$ and $\mu_i^{\text{red}}$ to update its strategies based on gradient play [8]. The optimization continues until nodes’ strategies converge to the solution of the game. More specifically, the players, states, strategies, actions, and payoff for the channel access game, $G$, are defined as follows:

- **Players**: The set of sensor nodes $\mathcal{N}$.
- **States**: The channel states $s_i(t)$.
- **Actions**: The action of each node is either to transmit or not to transmit, $a_i \in \{1, 0\}$, with transmission probability $p_i^{s_i(t)}$.
- **Strategy space**: $A_i = [p_i^{\min}, p_i^{\max}]$, and $\{p_i^{\min}, p_i^{\max}\} \in (0, 1)$.
- **Strategy**: $p_i^{s_i(t)} : s_i(t) \to \Delta(a_i)$.
- **Payoff**: The payoff $u_i$ is given by [8].
• Dynamics for strategy updates: Each node uses gradient play dynamics to update its own strategies as follows [8]:

\[ p_{s_i}(t+1) = \prod_{A_i} \left[ p_{s_i}(t) + \gamma \frac{\partial u_i}{\partial p_{s_i}(t)} \right], \]  

(10)

where \( \gamma \) is the step size, and \( \prod_{A_i} \) is the Euclidean projection of the transmission probability onto a strategy space \( A_i \).

To find the derivative of the stationary distribution, we establish the invertibility of the matrix \((\pi_i^{\text{red}} + 11^T - I)\) in the following lemma.

**Lemma 1.** The matrix \((\pi_i^{\text{red}} + 11^T - I)\) is an invertible matrix.

**Proof.** Assume that \((\pi_i^{\text{red}} + 11^T - I)\) is a non-invertible matrix. Then \( \exists \ v \in \ker(\pi_i^{\text{red}} + 11^T - I), \ v \neq 0 \) such that

\[ (\pi_i^{\text{red}} + 11^T - I)v = 0, \ v, 0 \in \mathbb{R}^4. \]  

(11)

Multiplying both sides of equation (11) by \( \mu_i^{\text{red}} \) as follows

\[ \mu_i^{\text{red}}(\pi_i^{\text{red}} + 11^T - I)v = 0, \]

\[ \mu_i^{\text{red}}\pi_i^{\text{red}}v + \mu_i^{\text{red}}11^Tv - \mu_i^{\text{red}}Iv = 0, \]

(12)

Substitute this in equation (11) yields the following

\[ \pi_i^{\text{red}}v + 11^Tv - Iv = 0, \]

\[ \pi_i^{\text{red}}v - Iv = 0, \]

(13)

\[ \pi_i^{\text{red}}v = v. \]

This implies that \( v = 1 \) is a right eigenvector of \( \pi_i^{\text{red}} \), with eigenvalue 1, and is unique by Perron-Frobenius theorem [9], which contradicts equation (12). Note that in the

\(^1\ker \) is the kernel or the Nullspace of the matrix.
previous steps, we use the assumption that $\pi_{\text{red}}$ is a positive and irreducible matrix, thus based on Perron-Frobenius theorem, $\mu_{\text{red}}$ is a positive and unique vector.

We proceed to find the derivative of the stationary distribution based on equation (7) as follows:

$$\frac{\partial \mu_{\text{red}}^i}{\partial p_{s}^{i(t)}} = -1^T \cdot (\pi_{\text{red}}^i + 11^T - I)^{-1} \cdot \frac{\partial \pi_{\text{red}}^i}{\partial p_{s}^{i(t)}} : (\pi_{\text{red}}^i + 11^T - I)^{-1},$$

(14)

which is based on a similar method for matrix inversion given in [10]. Based on equation (14), the individual entries of the derivative of the stationary distribution can be written as follows:

$$\frac{\partial \mu_{k}^i}{\partial p_{s}^{i(t)}} = -\mu_{k}^i \cdot \frac{\partial \pi_{\text{red}}^i}{\partial p_{s}^{i(t)}} : c_k,$$

(15)

where $c_k$ is the $k^{th}$ column of the $(\pi_{\text{red}}^i + 11^T - I)^{-1}$. The derivative of the payoff function is given by

$$\frac{\partial u_i(\mu_{10}^i(p), \mu_{11}^i(p))}{\partial p_{s}^{i(t)}} = (\alpha - \frac{\beta}{1 - \mu_{10}^i}) \cdot \frac{\partial \mu_{10}^i}{\partial p_{s}^{i(t)}} - \frac{\partial \mu_{11}^i}{\partial p_{s}^{i(t)}},$$

(16)

3.1. Nash Equilibrium

A solution concept to the channel access game $G$ is a Nash equilibrium $p^*$ in which the strategy vector $p_i^*$ of any node $i$ is its best response given the strategies $p_{-i}^*$ of the other nodes. This could be written as

$$p_i^* \in \text{BR}_i(p_{-i}^*).$$

In other words, no node has an incentive to unilaterally change its strategy vector at the point of the equilibrium, since this change will not increase its payoff. This can be defined for all nodes and strategies as follows:

**Definition 3.1.** Given the payoff function $u_i(\mu_{10}^i(p), \mu_{11}^i(p)) : p \rightarrow \mathbb{R}$ of the channel access game $G$, a strategy profile $p^*$ is said to be a Nash equilibrium if and only if

$$u_i(\mu_{10}^i(p_i^*, p_{-i}^*), \mu_{11}^i(p_i^*, p_{-i}^*)) \geq u_i(\mu_{10}^i(p_i, p_{-i}^*), \mu_{11}^i(p_i, p_{-i}^*)),$$

(17)
for any \( p_i \in A_i, i \in \mathcal{N} \). The following definition states the conditions that are required to verify a Nash equilibrium of the channel access game.

**Definition 3.2.** Given a twice continuously differentiable payoff function \( u_i : p \to \mathbb{R} \), a strategy profile \( p^* \) is said to be a local interior Nash equilibrium, if

1. The gradient of the payoff function is zero at this point, i.e.,
   \[
   \nabla u_i(\mu_{10}(p_i^*, p_{-i}^*), \mu_{11}(p_i^*, p_{-i}^*)) = 0, \forall i \in \mathcal{N}.
   \]

2. The hessian matrix, \( H(p^*) \) of the payoff function is negative definite/semi-definite at this point, \( \forall i \in \mathcal{N} \).

Furthermore, a local boundary Nash equilibrium is a strategy profile where,

\[
u_i(\mu_{10}(p_i^*, p_{-i}^*), \mu_{11}(p_i^*, p_{-i}^*)) \geq u_i(\mu_{10}(p_i, p_{-i}^*), \mu_{11}(p_i, p_{-i}^*))
\]

for any \( p_i^* \in [p_{\min}, p_{\max}], i \in \mathcal{N} \).

In this paper, we empirically verify the existence of a local boundary Nash equilibrium.

### 3.2. Channel Access Game Under Imperfect Information

In real systems, the information available to the nodes is imperfect due to the lack of communication among them. At node \( i \), this information is represented by a sequence of its own actions and the channel state observations \( s_i(t) \) instead of the actual joint state \( S(t) \). The solution concept of a Nash equilibrium point requires that agents are fully rational; which means that they play the best response for every possible history they observe. However, keeping track of every possible history is computationally intractable, which makes it necessary to seek another solution concept other than a Nash equilibrium, which considers the bounded rationality of the limited nodes. The Empirical Evidence Equilibrium (EEE), proposed in [11], is one of these solution concepts which we follow to define the equilibrium of the channel access game under imperfect information.

Consider a set of \( \mathcal{N} \) nodes playing a channel access game repeatedly in a real system \( R \). During this sequence of repeated interactions, node \( i \) observes \( O_i(t) = \)
and considers the channel state $s_i(t)$ as a signal which varies according to

$$s_i(t + 1) \sim f(s_i(t), a, S(t)).$$  \hfill (18)

To build a mockup for this signal, node $i$ assumes that these signals are generated by a Markov chain with a state $s_i(t)$, which is described by

$$s_i(t + 1) \sim m_i^1(s_i(t), s_i(t + 1)), \hfill (19a)$$
$$s_i(t + 1) \sim \tilde{\mu}_i(s_i(t)), \hfill (19b)$$

where $m_i^1(s_i(t), s_i(t + 1)) = s_i(t + 1)$ is a length-1-memory function, and $\tilde{\mu}_i(s_i(t))[s_i(t + 1)] = \mathbb{P}[s_i(t + 1)|s_i(t)]$ is the predictor of node $i$. In our analysis, we consider the pair $(m_i^1, \tilde{\mu}_i) = (m_i^1, \pi_i^{red})$ to be a mockup of node $i$. In the following, we use $\pi^{red}$ in order to refer to the predictors of all nodes.

**Example 6.** Suppose that the sequence of channel state observations is

$$(..., s_i(t - k), s_i(t - k + 1), ..., s_i(t), s_i(t + 1)) = (... , 00, 00, 01, 11),$$

then, the value of length-1-memory function is given by

$$m_i^1(z_i(t), s_i(t + 1)) = m_i^1(01, 11) = 11.$$  \hfill (20)

**Algorithm** explains the steps that nodes use to find the optimal parameters of their mockups. In this algorithm, the nodes implement their strategies $p$ in the real system $R$ and use their channel state observations, $s_i(t)$, to formulate a depth-1 consistent predictor. Then, the nodes use their predictors $\tilde{\mu}$ to update their strategies $p$ according to the gradient play dynamic. Given the updated strategies, the nodes switch back to the real system $R$ to compute their consistent predictors. The pair $(p, \pi^{red})$ is $m^{1}$-EEE (empirical evidence equilibrium) if it satisfies the conditions defined below:

**Definition 3.3.** Let $(m_i^1, \pi_i^{red})$ be a mockup and $p_i$ be the strategy vector of node $i$. The predictor $\pi_i^{red}$ is $(p, m_i^1)$ consistent with (18) if

$$\pi_i^{red}(s_i(t))[s_i(t + 1)] = \lim_{t \to \infty} \mathbb{P}_{R, p}[s_i(t + 1)|s_i(t)].$$  \hfill (20)
Definition 3.4. The strategy vector $\mathbf{p}_i$ is optimal for $u_i$ with respect to the mockup $(m^1_i, \pi^\text{red}_i)$ if for all nodes $i \in \mathcal{N}$, and all $\mathbf{p}, s_i$,

$$u_i(\mathbf{p}_i, \mathbf{p}_\sim i(\pi^\text{red}_i(s_i))) \geq u_i(\mathbf{p}_i, \mathbf{p}_\sim i(\pi^\text{red}_i(s_i))).$$  \hspace{1cm} (21)

In the following definition, we state the conditions required to verify the EEE equilibrium:

Definition 3.5. The pair $(\mathbf{p}, \pi^\text{red})$ is $m^1$-EEE if the following two conditions hold for all nodes $i \in \mathcal{N}$:

1. Strategy $\mathbf{p}_i$ is $u_i$ optimal with respect to $(\pi^\text{red}_i, m^1_i)$.

2. Predictor $\pi^\text{red}_i$ is $(\mathbf{p}, m^1_i)$ consistent with (18).

Proposition 1. The pair $(\mathbf{p}, \pi^\text{red})$ of the channel access game in the imperfect information scenario is $m^1$-EEE.

Proof. To prove the existence of $m^1$-EEE, we verify the conditions in Definition 3.5. The optimality of the strategy vector is achieved by the convergence of the gradient play to the critical point of the payoff function of each node. Under assumption (1), the Markov chain built from $s_i(t)$ is ergodic and thereby guarantees that $p(s_i(t)) > 0, \forall s_i(t)$ as well as the uniqueness of the stationary distribution over the channel states, $\mu_{s_i}^{t_i}$. Thus, the consistency can be verified as follows:

$$\pi^\text{red}_i(s_i(t))|s_i(t+1)] = \lim_{t \to \infty} \mathbb{P}_{R,p}|s_i(t+1)|s_i(t)], \ \forall s_i$$

$$= \pi^\text{red}_i(s_i(t))|s_i(t+1)]$$  \hspace{1cm} (22)

where $\pi^\text{red}_i(s_i(t))|s_i(t+1)]$ is the current value of a predictor of node $i$, which is computed based on Algorithm 1. This means that if the strategies of the nodes converge to their optimal estimated value, the entries of the transition matrix are the predictors of the next state $s_i(t+1)$ conditioned on the current state $s_i(t)$. 

$\square$
Algorithm 1 Estimation of Node $i$ Predictor in the Case of Imperfect Information Scenario

1: Initialize $\pi_i^{\text{red}} = 0$, MaxItr = $10^4$, MaxEpochItr = 0

2: loop

3: for epoch = 1 to MaxEpochItr do

4: for $j = 1$ to MaxItr do

5: Implement $p_i^{s_i(t)} : s_i(t) \rightarrow \Delta(a_i)$ to estimate $\pi_i^{\text{red}}$

6: end for

7: $\forall i \in \mathcal{N}$ and for each $s_i(t+1) \in \{11, 10, 01, 00\}$

8: $p(s_i(t+1)|s_i(t)) = \pi_i^{\text{red}}(s_i(t))[s_i(t+1)] = \frac{\sum_{t+1}^{\text{MaxItr}}(s_i(t+1)|s_i(t))}{\sum_{t+1}^{\text{MaxItr}}(s_i(t)|s_i(t))}$

9: Compute $\frac{\partial u_i}{\partial p_i^{s_i(t)}}, \frac{\partial y_i^{s_i(t)}}{\partial p_i^{s_i(t)}}, \frac{\partial \pi_i^{\text{red}}}{\partial p_i^{s_i(t)}}$

10: $p_i^{s_i(t)}(t+1) = \prod_A [p_i^{s_i(t)}(t) + \gamma \frac{\partial u_i}{\partial p_i^{s_i(t)}}]$

11: $\pi_i^{\text{red}}(s_i(t))[s(t+1)] = \pi_i^{\text{red}}(s_i(t))[s_i(t+1)] = p(s_i(t+1)|s_i(t))$

12: end for

13: end loop
4. Throughput Analysis

In this paper, we design the system to reach the optimal solution that maximizes the network throughput in the saturation conditions where each node always has a packet to transmit. Let $T_i$ be the normalized throughput, which is the fraction of time that nodes use to transmit their packets successfully, defined as follows:

$$T_i = \frac{E(\text{Payload information transmitted in a slot time})}{E(\text{Length of a slot time})} = \frac{P_s E(p)}{P_{idle} \sigma + P_s T_s + P_c T_c},$$

(23)

where $E(P)$ is the average packet payload size; $T_s$ is the average time the channel is sensed busy because of successful transmission; $T_c$ is the average time during a collision; $\sigma$ is the duration of empty time slot; $P_c$ is the collision probability; $P_{idle}$ is the probability of idle time slot; and $P_s$ is the successful transmission probability. The values for $T_s$ and $T_c$ are computed as follows:

$$T_s = \frac{ph}{br} + \frac{mh + P}{dr} + SIFS + \frac{ph}{br} + ACKbr + DIFS + 2\delta,$$

$$T_c = \frac{ph}{br} + \frac{mh + P}{dr} + DIFS + \delta,$$

(24)

see Table 1 for these notations and their values in IEEE 802.11b standard. In the case of a network with full topology, the work of [3] analyzed the IEEE 802.11 DCF protocol as a two-dimensional discrete-time Markov chain which has been used to find the optimal transmission probabilities of the nodes by solving the following set of nonlinear equations:

$$P_c = 1 - (1 - p_i)^{(N-1)},$$

(25a)

$$p_i = \frac{2(1 - 2P_c)}{(1 - 2P_c)(w + 1) + P_c w (1 - (2P_c)^m)},$$

(25b)

where $p_i$ is the transmission probability of node $i$, $m$ is the number of retransmissions, and $w$ is the minimum contention window size. These equations have been derived and solved with the assumption that all nodes experience the same probability of collision. We extend this approach to study the throughput of the nodes connected in a
network with any topology, we call it “analytical” IEEE 802.11 CSMA/CA, by solving equation (25b) and the following one:

\[ P_c = 1 - \prod_{j \in N(i)} (1 - p_j). \]  

(26)

5. Numerical Performance Evaluation

We consider a network of nodes with a strategy space of \( A_i \in [0.002, 1) \), and we show the performance results for different settings of our proposed solution. In these simulations, we assume that the queue of each node is saturated whereas the case of a node with variable queue size is under investigation. The values for the parameters that we used to obtain the numerical results are summarized in Table 1. These values are specified in the IEEE 802.11b standard where \( \xi^* = 0.1625 \) [5]. In all simulations, we set value \( \gamma = 0.05 \).

5.1. Empirical Verification of the Nash Equilibrium Point of Memory-one Access Game

We carried out a simulation of the memory-one channel access game for a fully connected network of \( N = 4 \) nodes. Fig. 1 shows the payoff that each node achieves by changing one of its \( p_{i(t)} \) and keeping the others fixed at their Nash equilibrium values.
Specifically, we discretize the value of \( p_i(t) \in [0, 1] \) and we compute the corresponding value of the payoff. Note from this figure that the payoff of node \( i \) is maximized at the equilibrium points of \( p_i^{s_i(t)} \in p^* \), which verifies that these strategies have achieved a Nash equilibrium point.

5.2. Equilibrium Performance of Memory-one Access Game vs. Memoryless Access Game

Suppose that the nodes access the channel using their Nash equilibrium strategies. To test the efficiency of the Nash equilibrium of the memory-one games against the efficiency of the Nash equilibrium of the memoryless channel access games, we consider networks of different sizes (different numbers of nodes) and run the channel access game for \( 10^6 \) iterations. At each iteration, the nodes update their stationary distribution vector \( \mu_i^{\text{red}} \). In Figure 2, we compare the stationary distributions of the memoryless and the memory-one channel access games, and conclude that the memory-one strategies enhance cooperation among the nodes since they result in a lower probability of collision, \( \mu^{11} \), and a higher probability of successful transmission, \( \mu^{10} \), compared to the memoryless strategies.
Fig. 2: Performance evaluation of Nash equilibrium points of the memory-one and memoryless channel access games played over networks of different sizes.

5.3. Throughput Comparison of a Fully Connected Network

We consider a network of nodes and compare the throughputs corresponding to our proposed method, memoryless channel access game, the idle sense [5], and IEEE 802.11b DCF protocol [3]. The values for the parameters used in the calculations are summarized in Table 1.

Fig. 3: Throughput comparison.

Fig. 3 shows that the values of the aggregate throughput achieved by our design are higher than the ones achieved by the benchmark methods. Furthermore, Fig. 4 shows that memory-one strategies achieve a lower collision probability and a higher successful transmission probability compared to the benchmark methods.
5.4. Memory-One Channel Access Game over Fully Connected Networks with Imperfect Information Scenarios

We study the channel access game in the case of an imperfect information scenario where the nodes access the channel and compute their parameters from their observations according to Algorithm 1. Fig. 5 and 6 compare the strategies and the payoffs of two nodes that use memory-one strategies in the case of perfect and imperfect information scenarios. We verify the empirical evidence equilibrium in the case of imperfect information by the convergence of the nodes’ strategies to the Nash equilibrium corresponding to the perfect information case.

5.5. Memory-One Channel Access Game over Networks with Different Topologies: Perfect Information Scenarios

To study the effect of the network topology on the throughput that nodes achieve by using memory-one, memoryless strategies, and the "analytical" IEEE 802.11b CSMA/CA, we ran simulations for different number of nodes connected in a line topology in the perfect information scenario; see Fig. 7 and 8 for the corresponding results. These figures show that the throughputs achieved by different numbers of nodes depend on their locations in the network with the line topology, and they achieve better fairness by using memory-one strategies. Furthermore, these figures show a significant improvement in the throughput achieved by memory-one strategies over the "analytical" IEEE
Fig. 5: Conditional transmission probabilities based on $\pi_i^\text{red}$ estimation for two nodes using memory-one strategies.

Fig. 6: Payoffs of two nodes under imperfect information where each node uses its memory-one strategies to estimate its $\pi_i^\text{red}$.

802.11b, while Fig. 3 shows a lower improvement in the throughput between them. This is because the collision rate in a fully connected network, which has one collision set, is higher than that in a network with a line topology, which has more than one collision set. This shows that the performance of the “analytical” IEEE 802.11b is sensitive to the size of the network and that the memory-one strategies perform better.
for a network of a large size.

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Throughput (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 7: Throughput comparison of a line topology network. Left: Network with size = 6 nodes. Right: Network with size = 5 nodes.

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Throughput (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 8: Throughput comparison of the "analytical" and the simulated ad hoc IEEE 802.11b CSMA/CA networks of line topology. Left: Network with size = 4 nodes. Right: Network with size = 5 nodes.

To validate our analysis in Section 4, we simulated the same network using ns-3 network simulator [12]. In our calculations, we used $w = 32$ and $m = 7$. Fig. 8 shows the consistency in terms of the throughput between the "analytical" and the simulated IEEE 802.11b CSMA/CA based network.

6. Conclusion

We investigated a random channel access game where nodes compete to access the channel without explicit coordination among them. Due to the selfish behavior for the shared resource, we used game theory as an optimization method to find the optimal
resource sharing strategy among nodes. We proposed a memory-one channel access game and studied the game under perfect and imperfect network information scenarios, depending on the realistic application environments and network architectures. Under the assumption of perfect network information, we empirically verified the existence of Nash equilibrium point. We further showed that at this equilibrium point, nodes achieve a higher throughput compared to the throughput that they achieve at their Nash equilibrium points in the case of the memoryless game and the IEEE 802.11 DCF protocol. In the case of the imperfect information scenario, the game converges to an equilibrium point which we modeled based on the Empirical Evidence Equilibrium framework. We further applied the channel access games to different contention graph topology in order to study the impact of different network configurations on the transmission probabilities of nodes. Our findings suggest that under the memory-one channel access game, nodes are able to reduce the number of collisions and increase their successful transmissions compared to those in the memoryless channel access games and the IEEE 802.11 DCF protocol.

Acknowledgment

The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST).

References


[12] ns-3 Network Simulator
URL https://www.nsnam.org