Selfish Task-Driven Routing in Hybrid Networks

Yupeng Li[1] Haisheng Tan[2] Yongcai Wang[3] Zhenhua Han[2,4] Francis C.M. Lau[1]
[1] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong
[2] Department of Computer Science, Jinan University, Guangzhou, China
[3] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China
[4] Department of EEE, the South University of Science and Technology of China, Shenzhen, China

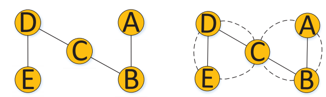
***Abstract*—In *Hybrid networks*, which synergistically mix together wired and wireless links to achieve flexible and reliable communication, it is particularly challenging to routing selfish tasks since each task wish to finish transmission as early as possible and its decision could have impacts on the others. In this paper, we investigate the problem to route a given set of selfish tasks in hybrid networks. Under a unified cost model, the competitive behaviors of selfish players are modeled as a noncooperative game. We show the game is ordinal potential, and the existence of a pure-Nash Equilibrium (pure-NE) is therefore guaranteed. We also design a routing scheme, called Selfish Task-Driven Routing (STaR), to achieve a pure-NE. Extensive simulations show that our scheme can not only efficiently converge to an equilibrium but also outperform other source routing protocols regarding the completion time and load balancing.**

I. Introduction

The gap between wired and wireless networks is narrowing because of the rapid developments in wireless technologies. Many situations exist today where blending together wired and wireless links could lead to much improved networking services. This paper studies such a hybrid network which combines the high-speed collision-free wired links with the flexible wireless links. Examples of hybrid networks are plentiful, including the ubiquitous Local Area Networks (LANs) where many user devices are connected wirelessly using Wi-Fi to a wired structure, and the emerging hybrid data center networks (HDCNs) [1], [2]. By capitalizing on the flexibility of the wireless links and the reliability of wired links, a hybrid network can offer many more routing options and a larger capacity. Fig. 1 illustrates the idea by a simplified hybrid network. The solid lines represent wired links, and the dashed ones are wireless links. Fig. 1(a) is a wired-only network with five nodes. From A to E, there is only one path: A→B→C→D→E. As for the hybrid network in Fig. 1(b), there are as many as 25 acyclic paths from A to E.

In a hybrid network, a node can be a router or a switch, an access point, a base station, a wireless device, and so on. Wired links can provide reliable as well as collision-free communication while the wireless links can provide flexible and more timely responses to communication requests. However, more available routing options provided by hybrid networks brings new challenges to the matter of route selection: 1) How

The first two authors are regarded as co-first authors. This work was supported in part by the Fundamental Research Funds for the Central Universities in China, and the National Natural Science Foundation of China Grant 61202360. Contact Haisheng Tan at thstan@jnu.edu.cn.



(a) (b)

Fig. 1: A hybrid network

to design a unified cost model for both the wired and wireless resources? 2) How to design routing rules based on the given cost model? 3) How to, in particular, route tasks that are selfish as they compete for the limited resources?

Selfish routing is difficult because different decisions of one task may have different impacts on the others. Take the hybrid network in Fig. 1(b) as an example. Suppose there are two tasks: the first is to route from A to E and the second from B to E. Assume that the first chooses wireless link A→C. Then, if the second task chooses wireless link B→C, the two transmissions may interfere with each other. However, if the second task chooses wired link B→C instead, both tasks can be accomplished without collision, which should lead to an overall better performance. Obviously, the problem can become much more complicated as the number of tasks increases.

In this work, we investigate the above routing problem in hybrid networks. As we just showed, different routing choices of one task will generate different impacts, i.e., different congestions and interferences, on the links. Therefore, the cost of choosing a particular link for a task is not static. In real applications, these impacts can be inferred by a central scheduler, such as the controller in a software defined network (SDN)[1]. We propose an interference matrix to describe the potential influence of one link on another, taking into account the protocol interference model and the CSMA/CA protocol. This matrix can be derived when the network topology and the wireless transmission powers are known. We consider the problem of routing a given set of selfish tasks in the hybrid network. Each task specifies a load (volume of data) to be delivered from a source to a destination, and each task desires to finish its transmission as early as possible.

For the routing game, we try to find a "stable" source routing strategy profile for each selfish task. Here, "stable" means no task can reduce its completion time by unilaterally changing its route. Our results are as follows:

[1]In this paper, the scheduler aims to assist each task to conduct the selfish routing.

- Given the selfish nature of tasks, the problem of source routing in hybrid networks is modeled as a noncooperative game. A cost system for paths is built based on the characteristics of the wired and wireless links.
- The proposed routing game is proven to be ordinal potential, and therefore possesses at least one pure-Nash Equilibrium (pure-NE). The price of anarchy is bounded by a measurement of homogeneity of the task size.
- By leveraging Joint Strategy Fictitious Play (JSFP) with inertia, we design an efficient routing scheme to achieve a pure-NE. Our routing scheme can also achieve good performance in terms of the completion time of the tasks and load balancing among the links.

To the best of our knowledge, this is the first work to investigate selfish task-driven routing game in hybrid networks where wired and wireless resources coexist.

Organization: The remainder of the paper is organized as follows. Related work is discussed in section II. Section III presents the system model. In Section IV, we formulate the routing game in hybrid networks and show that there is at least one pure-NE. In Section V, we give the design of a source routing scheme which can achieve the pure-Nash Equilibrium. We present numerical analysis and simulation results in Section VI. Finally, Section VII concludes the paper.

II. Related Work

In this section, we introduce the related work, including applications of hybrid wireless/wired networks, routing protocols in hybrid networks, and routing games.

A. Hybrid Wired/Wireless Networks

1) Home/Office Networking: It is now commonplace that the wired and wireless links coexist at home or in the office. However, currently, most of the routing schemes are rule-based, i.e., they set preference as to which of the wired Ethernet or Wi-Fi should be used when both are available.

2) Hybrid Mesh Networks: The hybrid mesh network is composed of a backbone (or infrastructure) network and an ad hoc network. The backbone is mainly established by wired links while the ad hoc network by wireless links. This architecture has been shown to be a better one for the next generation wireless networks [3].

3) Hybrid Data Center Networks: A hybrid data center network (HDCN) augments the traditional wired network with wireless links. It can provide flexible communications among the servers, whenever and wherever possible, as well as improve the overall network capacity [1], [2]. As stated above, with potentially the same level of bandwidth as the Ethernet, 60 GHz wireless technology [4] is considered a good candidate to support the added wireless links in hybrid networks.

B. Routing Protocols in Hybrid Networks

Very few works have investigated the routing problem in general hybrid wired/wireless networks. Some results are specific for a particular type of scenario, e.g., hybrid mesh networks.

Draves et al. [5] proposed a metric of route selection in multichannel networks, aiming at the problem of choosing a high-throughput path between a source and a destination. Raniwala et al. [6] focused on wireless mesh networks that use wired links to relay traffic among wireless access points, and proposed a routing protocol that required multi-interfaces on a node. Yoshida et al. [7] investigated how to determine whether to choose a route consisting of wireless links only. The most widely known source routing protocol is Dynamic Source Routing (DSR), which chooses a route on-demand when a transmitting node makes a request [8]. These works only focused on the networks integrating an infrastructure network and an ad hoc network but not a general hybrid wired/wireless network. For the hybrid data center, the problem to design an effective routing protocol remains open because the architecture of an ideal hybrid design is still being explored. Based on the above, there is a need for new routing protocols in generalized hybrid wired/wireless networks.

C. Routing Game

Some existing works investigated the selfish routing problem from a game-theoretical point of view [9]–[12]. Congestion games are a class of games proposed by Rosenthal [13]. Selfish routing games were studied in [10], including atomic and nonatomic ones. These classical studies on routing game did not care about possible interference among the links. Other papers studied the congestion game in a completely wireless environment. In [11], congestion is only caused by the neighbors, where wireless spectrum resources are shared by selfish users. The route-switching game in [12] studied a spectrum-mobility-incurred problem in which there is a trade-off between the channel-switching cost and the routing cost. They considered an exclusively wireless network environment without wired backbone. Further, each link with a specific channel was assumed to be available to at most one task throughout the routing process. In our work, we let any task have unrestricted use of the link resource, which makes the problem more challenging. *Existing routing games considered either a completely non-interfering environment (e.g., a wired network) or a completely wireless environment. Few works have investigated the routing game in hybrid wired/wireless networks.*

III. System Model

A. Hybrid Networks

A hybrid network can be represented by a potential graph $G = (V, E)$, where V is the set of nodes and E is the set of links which include both wired and potential wireless links. The potential available channel set is denoted as $\mathcal{C} = \{1, 2, \dots, C\}$. Similar to [6], we assume the number of antennae of each wireless node is enough to support all its potential links working on different available channels simultaneously. A potential wireless link exists between any two nodes that are within the wireless communication range of each other. Specifically, we denote E_X and E_Y as the sets of wired and wireless links respectively, and $E = E_X \cup E_Y$. Similar to other recent works on hybrid networks, we only consider *unicast* in the wireless communication, and assume that each wireless device has a stable power level. With a known power level, we can obtain the wireless transmission ranges and establish all the potential links. For simplicity, we assume each wireless node has the same power level.

In the wired environment, an edge $e_{i(p,q)} \in E_X$ denotes a link going from node N_p to N_q. Here, i is the link identity. For a bidirectional wired link, there are two parallel edges with opposite directions in the graph. In the wireless environment, an edge $e^c_{j(u,v)} \in E_Y$ from N_u to N_v is established if and only if N_v is within the transmission range of N_u on channel c, which is a potential link[2]. We assume that each node is equipped with multiple antennae and interfaces, so multiple wireless links can be established between two nodes. Two links are *opposite* if they differ from each other only by the direction, which are denoted as e and $\bar{e}$. For simplicity, we assume that all the links (both wired and wireless links) are with the same capacity. Our results are still valid when the links have different capacities.

B. Tasks

There are N concurrent tasks as indicated by $i \in \mathcal{N} = \{1, 2, \ldots, N\}$ in the hybrid network. Each task corresponds to a source-destination pair (s, t) and has a number of packets to transmit. We refer to the number of packets a task possesses as the task size. We assume that 1) each packet has the same size no matter which task it belongs to; 2) each task chooses one path to route all its packets. The goal of each task is to finish transmitting its data load as early as possible.

C. Interference Matrix

We use a matrix $\mathcal{I}$ to characterize the interference between any two links. For links $e_i, e_j \in E$, $\mathcal{I}(e_i, e_j)$ indicates how link e_i is interfered by link e_j.

In the wired environment, each link $e_X \in E_X$ will be interfered only by itself, which implies a congestion when the same wireline is used by a pair of connecting nodes. We set $\mathcal{I}(e_X, e_X) = 1$, and all other entries to zero. Specifically, we assume that opposite wired link pairs will not interfere with each other (working in the dual mode), denoted as $\mathcal{I}(e_X, \bar{e_X}) = 0$. In the wireless environment, only the links allocated with the same channel can interfere with each other. For the ones assigned with different channels, the corresponding entries of $\mathcal{I}$ are set to zero. We set $\mathcal{I}(e_i, e_i) = 1$, which denotes the congestion on this wireless link. As for the other entries, we determine the interference based on the protocol interference model. Recall that we only consider unicast in wireless communication.

In the protocol interference model, each entry of $\mathcal{I}$ is either 1 or 0. That is, for link pair $e_i, e_j \in E_Y$, $\mathcal{I}(e_i, e_j) = 1$ if and only if e_i and e_j are assigned the same channel and e_i is within the interference range of e_j, and $\mathcal{I}(e_i, e_j) = 0$ otherwise. Therefore, for a successful transmission on link e_i, we have

$$\sum_{e_j \in E \setminus \{e_i\}} \mathcal{I}(e_i, e_j) \cdot Z(e_j) = 0, \tag{1}$$

where $Z(e_j) = 1$ if and only if e_j is activated, i.e., included in at least one path that is routed by at least one task at a time, and $Z(e_j) = 0$ otherwise.

There is no interference between the wired links and wireless links. Note that the above matrix can be established when the geometry information and interference model are known, with the assumption that the power level of each wireless device is given.

D. Problem Definition

We consider the routing problem of N selfish tasks in a hybrid network. We assume a scheduler is available, which manages the information of the tasks, the network and the interference matrix. Specifically, the channel states, the network topology and information about the tasks, including the source-destination pairs and the loads to be transmitted, are collected at the scheduler. We are to design a centralized routing strategy for all the tasks, i.e., a strategy that no task can improve its performance (the completion time) without perturbing the others' decisions. Here, the completion time of a task is the time it takes to finish delivering all its load. We assume tasks will honestly follow the routing strategies. Our problem is formally defined as:

Problem 1. *Given a hybrid network G, channels $\mathcal{C}$, tasks $\mathcal{N}$ and the interference matrix $\mathcal{I}$, the problem is to design a routing scheme so that each task's* ***completion time*** *is minimized given the routing selections of the other tasks.*

IV. Task-Driven Routing Game Model

In this section, we define the routing game in a hybrid network with selfish tasks.

A. Link Cost

For any task $i \in \{1, 2, ..., N\}$, $\mathcal{P}_i$ is the set of its possible routing strategies. Only one path $p^k_i \in \mathcal{P}_i$ is chosen to route all its loads, where $k \in \{1, 2, ..., |\mathcal{P}_i|\}$. Let $\mathcal{P} = \bigcup_{i \in \mathcal{N}} \mathcal{P}_i$, which is the union of all the possible routing paths for the N tasks. The strategies of all the N tasks are within the space $\mathbf{P} = \mathbf{P}_1 \times \mathbf{P}_2 \times \cdots \times \mathbf{P}_N$, where a dimension $\mathbf{P}_i = \mathcal{P}_i$. We assume for each task i, from s_i to t_i there exists at least one feasible path composed of wired or wireless links.

We employ a two-dimensional matrix $\mathbf{X}^{\mathcal{N}}_{\mathcal{P}}$ to denote the path selection of each task, where $X^i_p = 1$ if the ith task chooses path p. Similarly, for any link $e \in E$, set $X^i_e = 1$ if this link is selected by the ith task, which means $e \in p$ and $X^i_p = 1$ ($p \in \mathcal{P}_i$). We denote $Q_i(e)$ as the loads of all the other tasks other than the ith task that will be transmitted on edge $e \in E$ during source routing. Note that $Q_i(e)$ is an important factor when choosing a path including link e, but not a measurement of the exact delay for task i's routing through this link. Recall that every task only chooses one path to transmit all of its

[2]We will use e^c_i and e_i for simplicity according to the context, all of which are literally equivalent to $e^c_{i(u,v)}$ in the wireless environment.

packets. α_i is the load that task i will transmit over e during its routing process. Thus, we have

$$Q_i(e) = \sum_{j \in \mathcal{N}_i} \sum_{p \in \mathcal{P} \ \& \ e \in p} X_p^j \cdot \alpha_j = \sum_{j \in \mathcal{N}_i} X_e^j \cdot \alpha_j, \quad (2)$$

where $\mathcal{N}_i := \mathcal{N} \setminus \{i\}$. The cost for task i to transmit its packets on link e is characterized as follows, which includes *the congestion cost* and *the interference cost*.

1) Congestion Link Cost: We allow routing multiple tasks on one link. Different tasks transmitting packets on the same link or on two opposite wireless links simultaneously will cause congestion.We utilize a cost function to formulate the congestion cost of task i on edge e, denoted as $D(i,e) = \lambda(\mathcal{I}(e,e)Q_i(e),\ \mathcal{I}(e,\bar{e})Q_i(\bar{e}))$. $\lambda(x,y)$ is a non-decreasing function of either x (with y fixed) or y (with x fixed). The congestion link cost reflects the potential queuing delay in wired and wireless communications.

2) (Wireless) Interference Link Cost: In the wireless environment, a pair of links using the same channel will interfere with one another. In the protocol interference model, we consider the CSMA/CA-like contention cost. In this contention protocol, one link that is interfered by some other links will contend for transmission since only one of them can transmit successfully at a time. We model the interference link cost as the expected contention delay. $F(i,e)$ denotes the interference link cost on e when task i selects it, which is defined as:

$$F(i,e) = \begin{cases} \mu(\mathcal{M}_i(e) - D(i,e)), & \forall e \in E_Y \\ 0, & otherwise, \end{cases} \quad (3)$$

where $\mathcal{M}_i = \mathcal{I} \cdot Q_i$ is a vector, $\mathcal{I}$ is the interference matrix we defined in Section III-C, and $\mu(\cdot)$ is a non-decreasing function.

We denote the total cost of the ith task routing on link e as $j(i,e)$, which can be calculated as follows.

$$j(i,e) = f(D(i,e), F(i,e)), \quad (4)$$

where f is also a non-decreasing function.

B. Path (routing) Cost

The strategy profile of all players is denoted as $A = \{p_1, p_2, \cdots, p_N\}$. Let A_i denote the strategy of player i, which actually is the path p_i it chooses. Set the strategy profile of all players other than player i as $A_{-i} = \{p_1, \cdots, p_{i-1}, p_{i+1}, \cdots, p_N\}$.

Under the profile A, for the ith task that chooses strategy A_i containing link $e_1(A_i), e_2(A_i), \cdots, e_k(A_i)$, the path cost $PC_i(A)$ is calculated as

$$PC_i(A) = PC_i(A_i, A_{-i}) = \sum_{e \in E} j(i,e) X_e^i, \quad (5)$$

where $A_i = p_i \in \mathcal{P}_i$ and $X_e^i = 1, \forall e \in A_i$. We evaluate the path cost $PC_i(A)$ by the expected end-to-end delay, which refers to the overall delay for loads of one task transmitted from the source to the destination. Without loss of generality, the packet size of each task is assumed to be identical.

C. Game Formulation

According to the cost formulation defined above, Problem 1 is transformed into Problem 2, as follows:

Problem 2. *Given a hybrid network G, channels $\mathcal{C}$, tasks $\mathcal{N}$ and the interference matrix $\mathcal{I}$, our problem is to design a routing scheme for all tasks such that each task's **path cost** is minimized given the routing selections of the other tasks.*

Clearly, the routing selection of each task depends on the decisions of the others. Thus, we formulate the problem as a routing game where each task acts as a player and selects its route selfishly to minimize its path cost. Recall that we assume there is a scheduler and each task has the information about the network and other tasks. We define the routing game in a hybrid network as $\mathcal{G} = \{G, \mathcal{N}, \mathbf{P}, PC\}$, where G is the potential communication graph, $\mathcal{N}$ is the N-player set corresponding to the tasks, and $\mathbf{P}$ is the strategy space. Each player $i \in \mathcal{N}$ has a strategy set $\mathcal{P}_i$ and a path cost function $PC_i : \mathbf{P} \to \mathbb{R}$. Next, we give the definition of the pure-Nash equilibrium for the routing game as follows.

Definition 1 (pure-Nash Equilibrium, pure-NE)**.** *A strategy profile A^* is called a pure-NE if for each player $i \in \mathcal{N}$*

$$PC_i(A_i^*, A_{-i}^*) \le PC_i(A_i, A_{-i}^*), \forall A_i \ne A_i^*,$$

which means that no player can reduce its cost by unilaterally changing its strategy at the equilibrium.

D. Existence of pure-NE

In this section, we show that the proposed routing game has a pure-NE by proving that it is an ordinal potential game.

1) Ordinal potential Game: Ordinal potential games are generalized from potential games which generalize a number of other games including the classical congestion game [14], [15]. The definition of the ordinal potential game is:

Definition 2 (Ordinal Potential Game)**.** *A finite (player and strategy) game is an ordinal potential game if and only if for some potential function $\phi : \mathbf{P} \to \mathbb{R}$,*

$$PC_i(A_i', A_{-i}) - PC_i(A_i'', A_{-i}) < 0$$
$$implies \ \ \phi(A_i', A_{-i}) - \phi(A_i'', A_{-i}) < 0,$$

for any player $i \in \mathcal{N}$, $A_{-i} \in \times_{j \ne i} \mathbf{P}_j$ and $A_i', A_i'' \in \mathcal{P}_i$.

The properties of ordinal potential games have been well studied [14]. The following one which gives us an approach to prove the existence of a pure-NE.

Lemma 1. *Every finite ordinal potential game possesses a pure-strategy Nash Equilibrium.*

2) Existence of pure-NE in the routing game: We show that our proposed routing game is a finite ordinal potential game, and thereby possesses a pure-NE.

Theorem 1. *Under the protocol interference model, given*

$$\begin{cases} j(i,e) = D(i,e) + F(i,e) \\ D(i,e) = \mathcal{I}(e,e)Q_i(e) + \mathcal{I}(e,\bar{e})Q_i(\bar{e}) \\ F(i,e) = \begin{cases} \mathcal{M}_i(e) - D(i,e), & \forall e \in E_Y \\ 0, & otherwise, \end{cases} \end{cases} \quad (6)$$

the routing game is a finite ordinal potential game.

Proof of Theorem 1. First of all, our proposed routing game has a finite number of players (tasks) and each player has a finite number of strategies (paths). So we have Claim 1.

Claim 1. *The routing game in hybrid networks is finite.*

Next, we prove that the game is an ordinal potential game. We denote one specific strategy profile (which corresponds to a point in space $\mathbf{P}$) as $A(a) = \{A_1(a), A_2(a), \cdots, A_N(a)\}$, where $A_i(a) = \{e | e \in A_i(a), X_e^i(a) = 1\}$ is the ith task's strategy. Similarly, let $A_{-i}(a)$ denote the same strategy profile excluding the player i, and we have $A(a) = (A_i(a), A_{-i}(a))$. Then, the path cost is denoted as $PC_i(A_i(a), A_{-i}(a))$.

Lemma 2. *Under the protocol interference model, there is an ordinal potential function in the routing game in hybrid networks, which can be* $\phi(a) = \sum_{i\in\mathcal{N}} \alpha_i PC_i(A(a))$.

Proof of Lemma 2. Suppose there are two strategy profiles $A(a)$ and $A(b)$ satisfying that

$$PC_k(A_k(a), A_{-k}(a)) < PC_k(A_k(b), A_{-k}(b)), \quad (7)$$

where $A_k(a) \neq A_k(b)$ and $A_{-k}(a) = A_{-k}(b)$.

Under the protocol interference model, we assume $\mathcal{I}(e, e') = \mathcal{I}(e', e)$ similar to [12]. According to the formulae in Theorem 1, we can rewrite the path cost of the kth task

$$PC_k(A) = \sum_{e\in A_k} j(k, e) = \sum_{e\in p}\sum_{e'\in E}\sum_{i\in\mathcal{N}_k} \mathcal{I}(e, e')\alpha_i X_{e'}^i. \quad (8)$$

We have

$$\begin{aligned}\phi(a) - \phi(b) &= \sum_{i\in\mathcal{N}} \alpha_i PC_i(A(a)) - \sum_{i\in\mathcal{N}} \alpha_i PC_i(A(b)) \\ &= \alpha_k PC_k(A(a)) - \alpha_k PC_k(A(b)) \\ &+ \sum_{i\in\mathcal{N}_k} \alpha_i PC_i(A(a)) - \sum_{i\in\mathcal{N}_k} \alpha_i PC_i(A(b)).\end{aligned} \quad (9)$$

Since $A_i(a) = A_i(b), \forall i \in \mathcal{N}_k$, we get

$$\begin{aligned}\phi(a) - \phi(b) &= \alpha_k \sum_{e\in E} X_e^k(a) \sum_{e'\in E}\sum_{j\in\mathcal{N}_k} \mathcal{I}(e, e')\alpha_j X_{e'}^j(a) \\ &- \alpha_k \sum_{e\in E} X_e^k(b) \sum_{e'\in E}\sum_{j\in\mathcal{N}_k} \mathcal{I}(e, e')\alpha_j X_{e'}^j(b) \\ &+ \alpha_k \sum_{e'\in E} X_{e'}^k(a) \sum_{e\in E}\sum_{i\in\mathcal{N}_k} \mathcal{I}(e, e')\alpha_i X_e^i(a) \\ &- \alpha_k \sum_{e'\in E} X_{e'}^k(b) \sum_{e\in E}\sum_{i\in\mathcal{N}_k} \mathcal{I}(e, e')\alpha_i X_e^i(b) \\ &= 2\alpha_k[PC_k(A(a)) - PC_k(A(b))] < 0.\end{aligned} \quad (10)$$

According to Inequality 7 and Definition 2, Lemma 2 is proved. □

Based on the above, Theorem 1 is proved. □

Therefore, we conclude that there is at least one pure-NE in our proposed routing game in hybrid networks.

V. Selfish Task-Driven Routing Scheme

In this section, we propose a source routing scheme in hybrid wired/wireless networks, called the Selfish Task-Driven Routing (STaR) scheme. We first adopt a learning scheme, joint strategy fictitious play (JSFP) with inertia [16], to find a pure-NE of our game, whose existence is proved in Section IV. The learning scheme performs well especially in large-scale ordinal potential games [14]. After finding the pure-NE, we present the pure Price of Anarchy.

A. Routing Scheme

Fig. 2 shows the framework of the scheme. The above three blocks show operations in the outer layer. In each iteration, the scheme gathers information of each task with the scheduler. The information includes the corresponding source-destination pair and the size of each task. Then the learning algorithm JSFP with Inertia is called to achieve a pure-NE. Whenever a pure-NE is reached, the scheduler returns the routing selection in the strategy profile for all tasks. Finally, tasks do source routing following the selected routes. This scheme assigns each selfish task fairly with a route, and we can see in the the next section load balancing is also achieved.

B. JSFP with Inertia

The lower part of Fig. 2 illustrates JSFP with inertia. It first samples a profile of route selections for all tasks (Initialization), and then updates the predicted cost by recursion and selects either the strategy in last time slot (with inertia) or from a set of candidates with some probability (Update). JSFP needs minimal computational support [16]. It requires neither to track the empirical frequencies nor to compute the expected costs of joint strategies of the other players.

Algorithm 1 is the framework of using JSFP with Inertia, which has two stages, the Initialization (see Algorithm 2) and the Update (see Algorithm 3). We have:

Theorem 2. *In our proposed routing game, the strategy profile $A(t)$ generated by Algorithm 1 converges to a pure-NE with a high probability.*

The convergence is almost for certain if each route of any task has a unique cost. This property of JSFP with inertia was proved in [16].

The notations in the algorithms are as follows. Time is divided into time slots. We use iteration interchangeably for time slot since each task will select a strategy according to the strategy profile at the last time slot. The strategy profile at time slot t is $A(t)$. $A_i(t)$ and $A_{-i}(t)$ are the strategy selected by the ith task and the strategy profile selected by all other tasks at time slot t, respectively. $I\{\cdot\}$ (Step 3 and 4 in Algorithm 2) is an indicator function. $I\{A(t) = \bar{A}\} = 1$ means the joint strategy profile selected by all tasks at time slot t is $\bar{A}$, and $I\{A(t) = \bar{A}\} = 0$ otherwise. $z_{-i}^{\bar{A}_{-i}}(t)$ is the percentage of times at which tasks other than i have selected the joint strategy profile $\bar{A}_{-i} \in \mathcal{P}_{-i}$, which is defined as $z_{-i}^{\bar{A}_{-i}}(t) := \frac{1}{t}\sum_{\tau=0}^{t-1} I\{A_{-i}(\tau) = \bar{A}_{-i}\}$. $z_{-i}(t)$ (Step 3 in Algorithm 1) is the empirical frequency vector formed by

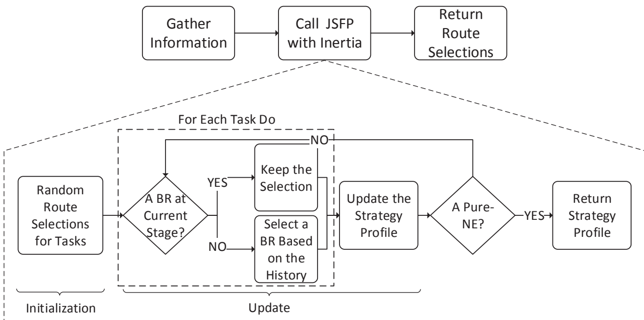


Fig. 2: The Framework of JSFP with Inertia Based Selfish Task-Driven Routing (STaR) Scheme in Hybrid Networks.

Algorithm 1: Framework of JSFP with Inertia.

Input: Task ID i, $\{\mathcal{P}_i\}_{i=1}^{N}$, K
Output: A sequence of strategies for the ith task to act on at t

Call Algorithm 2: Initialization;
repeat
  $t = t + 1$; Update $z_{-i}(t)$;
  Call Algorithm 3: Update;
until $k = K$;

$\{z_{-i}^{\bar{A}_{-i}}(t)\}_{\bar{A}_{-i} \in \mathcal{P}_{-i}}$. $\bar{PC}_i^{\bar{A}_i}(t)$ (Step 2 in Algorithm 3) is the average cost for the ith task selecting strategy $\bar{A}_i$ which is given by $\bar{PC}_i^{\bar{A}_i}(t) := \frac{1}{t}\sum_{\tau=0}^{t-1} PC_i(\bar{A}_i, A_{-i}(\tau))$.

The algorithm is designed for each task, helping it to determine its own route for source routing while considering the selfishness of all other tasks. Note that each task is required to obtain the strategy profile of other task in each iteration, so global synchronization is needed. We describe the algorithm briefly, as follows.

At the stage of initialization (Algorithm 2), each task selects a best-response strategy with the assumption that other tasks randomly selects a path. At the stage of update (Algorithm 3), in the tth ($t \geq 1$) iteration, a task, say i, will select a path based on two cases. The first case is that task i will stay on its strategy in the last iteration if it is also a best response in the current iteration (Line 8). The second case is that task i will select a strategy according to a probability distribution, $\omega_i(t)\rho_i(t) + (1 - \omega_i(t))v^{A_i(t-1)}$, if the strategy in the last iteration is not a best response in the current iteration (Line 10). $\omega_i(t)$ is the probability of the second case, while $(1 - \omega_i(t))$ is the probability of the first. $\rho_i(t)$ can be any probability distribution whose support is contained in $BR_i(z_i(t))$, the set of best responses for task i. $v^{A_i(t-1)}$ denotes a binary vector of size $|\mathcal{P}_i|$, in which only the entry corresponding to the strategy in iteration $t-1$ is set to 1.

C. Pure Price of Anarchy

Selfishness in our proposed routing game may influence the global performance in hybrid networks, which is studied through the analysis of pure-PoA (i.e., the PoA under a pure

---

Algorithm 2: Framework of JSFP with Inertia: Initialization.

Input: Task ID i, $\{\mathcal{P}_i\}_{i=1}^{N}$
Output: A strategy for the ith to act on at $t = 0$

$t = 0$, $k = 0$, $\bar{PC}_i^{\bar{A}_i}(0) = 0$;
Uniformly randomly select $\bar{A}_{-i} \in \mathcal{P}_{-i}$;
$I\{A_{-i}(0) = \bar{A}_{-i}\} = 1$;
$I\{A_{-i}(0) = A_{-i}\} = 0, \forall A_{-i} \neq \bar{A}_{-i} \in \mathcal{P}_{-i}$;
for $\forall \bar{A}_i \in \mathcal{P}_i$ **do**
  Calculate $\bar{PC}_i^{\bar{A}_i}(0)$;
for $\forall \bar{A}_i \in \mathcal{P}_i$ **do**
  if $\bar{PC}_i^{\bar{A}_i}(0) = \min_{A_i' \in \mathcal{P}_i} \bar{PC}_i^{A_i'}(0)$ **then**
    $\tilde{A}_i(0) = \bar{A}_i$;
    Include $\tilde{A}_i(0)$ in the set $BR_i(z_i(0))$;
Uniformly randomly select a strategy $\tilde{A}_i \in BR_i(z_{-i}(0))$ and set $A_i(0)$ to be that strategy;
return $A_i(0)$;

Algorithm 3: Framework of JSFP with Inertia: Update.

Input: Task ID i, t, k, $A_{-i}(t-1)$, $\bar{PC}_i^{\bar{A}_i(t-1)}, \forall \bar{A}_i \in \mathcal{P}_i$
Output: A strategy $A_i(t)$ for the ith to act on at t

1 **for** $\forall \bar{A}_i \in \mathcal{P}_i$ **do**
2   Calculate: $\bar{PC}_i^{\bar{A}_i}(t) = \frac{t-1}{t}\bar{PC}_i^{\bar{A}_i}(t-1) + \frac{1}{t}PC_i(\bar{A}_i, A_{-i}(t-1))$;
3 **for** $\forall \bar{A}_i \in \mathcal{P}_i$ **do**
4   **if** $\bar{PC}_i^{\bar{A}_i}(t) = \min_{A_i' \in \mathcal{P}_i} \bar{PC}_i^{A_i'}(t)$ **then**
5     $\tilde{A}_i(t) = \bar{A}_i$;
6   Include $\tilde{A}_i(t)$ in the set $BR_i(z_{-i}(t))$;
7 **if** $A_i(t-1) \in BR_i(z_{-i}(t))$ **then**
8   $A_i(t) = A_i(t-1)$; $k = k + 1$;
9 **else**
10   Select a strategy $A_i(t)$ based on the probability distribution: $\omega_i(t)\rho_i(t) + (1 - \omega_i(t))v^{A_i(t-1)}$;
11   $k = 0$;
12 **return** $A_i(t)$;

---

strategy game). An objective function is given in Definition 3, which characterizes the global cost and inherently reflects the expected latency that a hybrid network suffers. The pure-PoA is also defined.

Definition 3 (Objective Function, obj)**.** *The objective function under a strategy profile* $A(a)$ *is* $obj(A(a)) := \sum_{i \in \mathcal{N}} PC_i(A(a))$.

Definition 4 (Pure Price of Anarchy, pure-PoA)**.**

$$\textit{pure-PoA} := \frac{\max_{A^*} obj(A^*)}{\min_{A(a)} obj(A(a))}$$

where $\max_{A^*} obj(A^*)$ *corresponds to the worst objective function among all the pure-NEs.*

Based on the above, we have the following theorem:

Theorem 3. *The pure-PoA in the proposed routing game in hybrid networks is bounded by* $\max_{i,j \in \mathcal{N}} \frac{\alpha_i}{\alpha_j}$, *which reflects the homogeneity of the traffic load demands of each task.*

Proof of Theorem 3. Under the protocol interference model, we can derive

$$(\min_i \alpha_i) \cdot obj(A(a)) \leq \phi(a) \leq (\max_i \alpha_i) \cdot obj(A(a)).$$

Let ϕ^* be the ordinal potential function of a pure-NE strategy profile. Then

$$\phi^* \leq \phi(a) \implies \text{pure-PoA} \leq \max_{i,j \in \mathcal{N}} \alpha_i / \alpha_j.$$

Therefore, our theorem is proved. □

VI. Evaluation

We evaluate the performance of the proposed JSFP with inertia based Selfish Task-Driven Routing (STaR) scheme in the ns3 simulator.

We conducted simulations in two different scenarios of hybrid networks deployed in a 50m×50m region: a) 10 nodes with 15 wired links, b) 15 nodes with 25 wired links. The nodes and wired links are generated randomly. As stated in Section III, each node is equipped with adequate network interfaces and antennae for transmissons on its available channels simultaneously, and each node has the same wireless transmission power level. There are totally 3 orthogonal channels available in the network. Under the protocol interference model, the transmission range and interference range of each node are set as 10m and 11m, respectively. A wireless link is established between two nodes if and only if they are within the wireless transmission range of each other. The capacity of wired links and the wireless links are set to 10 Mbps, each of which has a propagation delay of 2 us. The task number ranges from 25 to 50. The source and destination of each task are randomly selected. The task sizes are also randomly chosen based on a Gaussian distribution with a mean value of 20 MB, a standard deviation of 10/3 MB, and restricted between 10 MB and 30MB. In each case, a scheduler selects a path for each task according to the STaR scheme. In the JSFP with inertia, $\omega_i(t)$ is set as 0.5 for any task at any time, and $\rho_i(t)$ is set to be a uniform distribution.

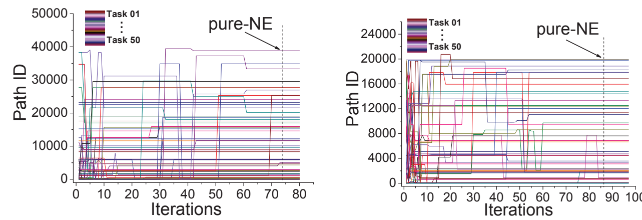
A. Evaluation of Routing Selection

In this section, we evaluate the efficiency of our proposed routing selection scheme STaR, and the impact of the parameters are also analyzed.

1) Efficiency: We first give the definition of the near-Nash equilibrium or an ϵ-Nash equilibrium:

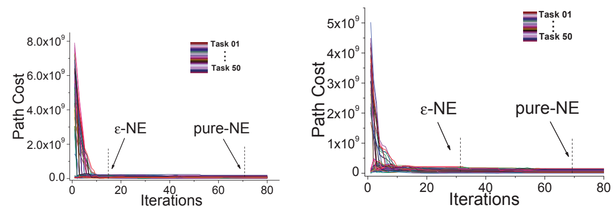
Definition 5 (near-Nash Equilibrium, ϵ-Nash Equilibrium, ϵ-NE)**.** *A strategy profile* $A^\dagger$ *is called an* ϵ*-Nash Equilibrium if for all players* $i \in \mathcal{N}$

$$PC_i(A_i^\dagger, A_{-i}^\dagger) \leq PC_i(A_i, A_{-i}^\dagger) + \epsilon, \forall A_i \neq A_i^\dagger,$$



(a) The 1st scenario. (b) The 2nd scenario.

Fig. 3: The evolution of the routing selections of each task before reaching a pure-NE (task # =50).



(a) The 1st scenario. (b) The 2nd scenario.

Fig. 4: The evolution of the path costs of each task before reaching the ϵ-NE and pure-NE (task # = 50).

which means that no player can reduce its cost by ϵ *through unilaterally changing its strategy at the* ϵ*-NE.*

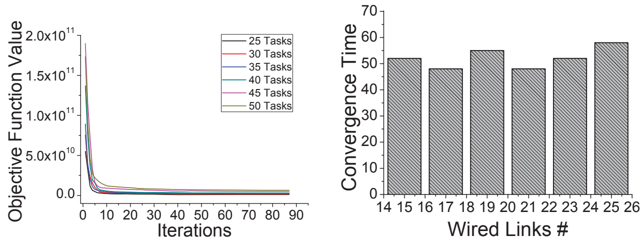
Fig. 3 illustrates the evolution of the routing selections of the 50 tasks before reaching a pure-NE. STaR takes on average about 70 and 85 iterations to achieve a pure-NE in the two scenarios respectively. We can see in Fig. 4 that the path costs of the tasks fluctuates frequently in the first 20 iterations on average and levels off after that. This implies, by setting a value of ϵ, we can quickly obtain an ϵ-Nash equilibrium which is of great importance in large-scale routing games. Based on the above observations, we can conclude our STaR scheme converges to an equilibrium effectively and efficiently.

2) Impact of Parameters: We now discuss the influence of two key parameters on the performance of STaR. ***1) the number of tasks:*** We can conclude from Fig. 5(a) that the evolutions of objective functions (refer to Definition 3) in different cases are almost identical; ***2) the number of wired links:*** Fig. 5(b) illustrates that when the number of wired links changes from 15 to 25, the time to reach an pure-NE does not have an obvious variation.

B. Comparing STaR with DSR

In this section, we compare the performance of our proposed STaR scheme with the popular routing scheme, dynamic source routing (DSR). Two key metrics are considered, the completion time of each task and the load balancing of the links. Since similar results are achieved in the two scenarios, we only illustrate the second scenario here.

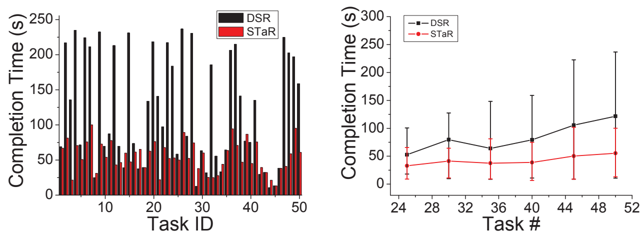
1) Completion Time: Fig. 6(a) shows the completion time of 50 tasks in the second scenario by adopting the DSR and the STaR scheme. We can find that almost all the selfish tasks are completed in a much earlier time in our STaR scheme.



(a) The impact of task number: the evolution of objective function

(b) The impact of wired link number: 15 nodes

Fig. 5: The impact of parameters.



(a) Completion time of the 50 tasks

(b) The impact of task number

Fig. 6: Completion time.

We illustrate in Fig. 6(b) the maximal (the higher endpoint of each vertical segment), the minimal (the lower endpoint), and the average (the dot at the middle of each vertical segment) completion time among all the tasks when the task number changes. Compared with DSR, STaR achieves a dramatic decrease in the average and maximal completion time. For the minimal completion time, STaR performs better in only nearly half of the cases, since DSR selects a route with approximately minimal latency for all tasks.

2) Load-Balancing: A Chebyshev's Sum Inequality based metric is used to measure the load-balancing performance of the links similar to [17], which is given by $\theta = \frac{\sum_{e \in E} L(e)^2}{|E| \sum_{e \in E} L(e)^2}$, where $L(e)$ is the load running on the link e. It is easy to verify that θ is not larger than 1, and the higher θ is, the better the performance in load balancing. Fig. 7 shows that STaR achieves a dramatic increase in the load-balancing based on the metrics θ compared with DSR.

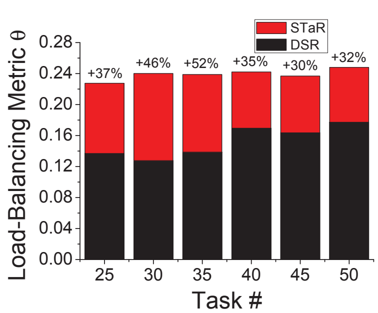


Fig. 7: Evaluation of load-balancing.

VII. Conclusion

In this paper, we addressed the selfish routing problem in hybrid wired/wireless networks. The joint routing process of the selfish tasks was formulated as a noncooperative game. We proved that the game was ordinal potential, and the existence of pure-Nash Equilibrium was therefore guaranteed. We also designed a centralized selfish task-driven source routing scheme, called STaR, leveraging the Joint Strategy Fictitious Play with inertia. Extensive simulations showed that the scheme not only can find a routing strategy profile for all tasks efficiently but also outperform the existing routing scheme regarding the completion time and load balancing. This work opens up several avenues for future research. One is to consider a statistical model to predict the current strategy profile of other tasks, so that each task does not need to receive global information from the scheduler at each iteration. We can also investigate how to improve the wireless capacity by power control in hybrid networks.

References

[1] D. Halperin, S. Kandula, J. Padhye, P. Bahl, and D. Wetherall, "Augmenting data center networks with multi-gigabit wireless links," in *SIGCOMM*, 2011.

[2] N. Hamedazimi, Z. Qazi, H. Gupta, V. Sekar, S. R. Das, J. P. Longtin, H. Shah, and A. Tanwer, "Firefly: a reconfigurable wireless data center fabric using free-space optics," in *SIGCOMM*, 2014.

[3] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer networks*, vol. 47, no. 4, pp. 445–487, 2005.

[4] J.-Y. Shin, E. G. Sirer, H. Weatherspoon, and D. Kirovski, "On the feasibility of completely wirelesss datacenters," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 5, pp. 1666–1679, 2013.

[5] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *MOBICOM*, 2004.

[6] A. Raniwala, K. Gopalan, and T.-c. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," in *MC2R*, 2004.

[7] M. Yoshida and T. Miki, "A routing protocol for wired-gum-wireless ad hoc networks," in *APCC*, 2005.

[8] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile computing*.

[9] N. Nisan, "Algorithms for selfish agents," in *STACS 99*. Springer, 1999, pp. 1–15.

[10] T. Roughgarden and É. Tardos, "How bad is selfish routing?" *Journal of the ACM (JACM)*, vol. 49, no. 2, pp. 236–259, 2002.

[11] R. Southwell and J. Huang, "Convergence dynamics of resource-homogeneous congestion games," in *GameNets*, 2012.

[12] Q. Liang, X. Wang, X. Tian, F. Wu, and Q. Zhang, "Two-dimensional route switching in cognitive radio networks: A game-theoretical framework," *IEEE Transactions on Networking*, vol. PP, 2014.

[13] R. W. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *International Journal of Game Theory*, vol. 2, no. 1, pp. 65–67, 1973.

[14] D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124–143, 1996.

[15] G. Christodoulou and E. Koutsoupias, "The price of anarchy of finite congestion games," in *STOC*, 2005.

[16] J. R. Marden, G. Arslan, and J. S. Shamma, "Joint strategy fictitious play with inertia for potential games," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 208–220, 2009.

[17] Y. Wang, Y. Wang, H. Tan, and F. C. Lau, "Maximizing network lifetime online by localized probabilistic load balancing," in *ADHOC-NOW*, 2011.