A decentralized load balancing algorithm for heterogeneous wireless access networks
Guido Oddi, Antonio Pietrabissa, Francesco Delli Priscoli, Vincenzo Suraci
Department of Computer, Control and Management Engineering “Antonio Ruberti”
University of Rome “La Sapienza”, Rome, Italy
{oddi, pietrabissa, dellipriscoli, suraci}@dis.uniroma1.it

Abstract

Modern mobile devices (e.g., laptops, smartphones, tablets, etc.) are capable to run a wide set of different applications, with increasing throughput demands. At the same time, those devices are equipped with a set of heterogeneous interfaces to wireless access networks (e.g., Wi-Fi, UMTS/LTE-A, WiMAX, etc.). This paper proposes a decentralized load balancing algorithm based on game theory (and in particular on the concept of Wardrop equilibrium), which, according to the feedback gathered from the environment, is able: (i) to balance the traffic among the available wireless access technologies, with the aim of increasing the overall throughput and, consequently, to satisfy the needs of the users; (ii) to be reactive to possible network status changes (e.g., increasing of packet loss probability, link failures, etc.), by performing technology handover. The simulations show the higher performances obtained by the proposed algorithm, in terms of application throughput, compared to two approaches: one which statically choose a unique technology and the other one which dynamically choose the technology with the current smallest packet loss.

1 Introduction

Modern mobile devices (e.g. laptops, smartphones, tablets, etc.) are capable to run a wide set of different applications, with increasing throughput demands (e.g., video streaming, real-time video conferences, on-line gaming, etc.). At the same time, those devices are equipped with several heterogeneous interfaces to wireless access networks (Wi-Fi, UMTS/LTE-A, WiMAX, etc.), which try to make the users always connected to the Internet. Protocols running on top of the wireless connections are not currently capable of jointly using the available technologies to increase the throughput of the users’ applications. Some approaches are developed in home scenarios ([1],[2],[10]): in this kind of networks, there is a centralized entity (the home gateway), in charge of distributing the traffic over the available technologies to increase the overall throughput. However, those approaches are not applicable in a mobile scenario, in which the wireless access networks belong to different administrative entities and no centralized resource manager could be foreseen, in this context. As a consequence of the previous discussion, in the scenario considered in this paper, only decentralized or distributed approaches could be applicable and must be designed by network engineers. This paper presents a novel decentralized load balancing algorithm based on game theory (and in particular on the concept of Wardrop equilibrium), which, according to the feedback coming from the environment, is able to balance the traffic among the available wireless access technology, to increase the overall throughput and, consequently, to satisfy the needs of the users. Moreover, the proposed feedback-based approach makes the algorithm aware of the current condition of the network and act as a consequence (i.e., performing handover among technologies).

The paper is organized as follows. Section 2 introduced the main concept, the scenario considered in this paper, the load balancing problem and a reference architecture, which enables the deployment of load balancing in heterogeneous wireless networks. Section 3 illustrates the load balancing model as well as the concept of Wardrop equilibrium. Section 4 describes the proposed algorithm, an adaptation of the approach proposed in [5]. Section 5 reports a set of simulations carried out to validate the proposed algorithm. Finally, Section 6 draws the conclusions.

2 Main concept

This section shows the concept of load balancing over a set of heterogeneous technologies, in particular wireless technologies. The proposed algorithm is applicable in decentralized environments such as the one considered in this paper. Section 2.1 introduces the concept of load balancing and the scenario considered in this paper. Section 2.2 introduced an architectural enabler to allow the deployment of such kind of algorithms, namely, the Multi-Connection Transport Layer.

2.1 Scenario and the load balancing problem

We consider, in this paper, a scenario similar to the one illustrated in Figure 1. A set of \( n \) users is able to connect either alternatively or simultaneously to a set of heterogeneous wireless technologies, through the related Access Points (APs). Figure 1 shows two technologies
(Wi-Fi and UMTS) but the proposed approach could be easily extended to handle a wide multiplicity of available radio technologies (e.g., WiMAX, LTE-A, Satellite, etc.). The wireless technologies allow the connected users to access to the Internet.

**Figure 1** An example of scenario.

The objective of this paper is to present a load balancing algorithm applicable to such kind of scenario. The main characteristic of load balancing algorithms is the ability to make use of a set of multiple (heterogeneous) technologies to transmit the traffic flows and to increase the overall throughput. Two typologies of load balancing are studied in literature: per-flow load balancing (see for instance [1], [11]) and per-packet load balancing (see for instance [2] or [12], from an architectural point of view). The first approach imposes that a flow cannot be split over multiple technologies: thus, a flow is completely assigned to a specific technology. The second approach relaxes such assumption and allows flows to be divided onto multiple technologies. The algorithms of the per-flow family are characterized by a better simplicity of implementation in real networks but yield worse results than the per-packet algorithms. The algorithm presented in this paper belongs to the per-packet family. The main drawback of per-packet algorithms is that, if the load balancing procedure is carried out below the TCP layer, the difference in delays experienced by the multiple technologies used by a single TCP connection may lead to an increase of retransmissions and the failure of the whole load balancing algorithm. To avoid such shortcomings, the load balancing must be performed over the transport layer. The enabler is the so-called Multi-Connection Transport Layer, described in the following section.

### 2.2 The Multi-Connection Transport Layer

As described in the previous section, the Multi-Connection Transport Layer is the enabler of the load balancing algorithms belonging to the per-packet family. In fact, it allows to overcome the shortcomings of the combination of load balancing below IP layer with the TCP layer. The Multi-Connection Transport Layer acts as a control plane allowing the use of multiple transport connections (one for each used technology connected to the mobile device). An example of node architecture (in terms of placement in the internet stack) could be the following (see [3] for a similar approach):

**Figure 2** The Multi-Connection Transport Layer

Each interface is associated with an IP address, assigned by the DHCP server inside the Access Point, which gives connectivity (for instance, the Wi-Fi AP, the WiMAX AP, the UMTS AP, etc.). For that reason, since the management of the two segments is different, separate and difficult to integrate, the convergence among technologies is possible only above the Network (IP) Layer (differently from [4]). The main idea of the Multi-Connection Transport Layer is to generate a separate connection (i.e., UDP or TCP connection) for each interface used by the mobile device, and control the amount of packets (bandwidth) sent over each connection, according to the underlying links status (in particular the packet loss sensed on each separate radio technology, as clearly defined below). The link status is delivered at transport layer by means of the Cross Layer module.

**Figure 3** Interfacing with external content servers.

The presence of the Multi-Connection Transport Layer forces also the counterpart, communicating with the mobile device, to establish multiple connections with the mobile device and, thus, to run the Multi-Connection Transport Layer. This is done, for instance, in [3], in which the servers (e.g., streaming or file servers) are compatible with the new transport layer.

Another idea, in order to make the proposed approach feasible for each possible entity interacting with the mobile device, is to deploy a set of gateway servers (managed by a third party which could make business from this opportunity) in charge of interfacing with the devices, starting and handling a set of transport layer connections and acting as a gateway towards the third parties (e.g., providing streaming services, video
conferencing, etc.). Such concept is illustrated in Figure 3. The system architecture is also valid in case the communication is between two mobile devices, belonging to different heterogeneous networks, able to split traffic among two different interfaces. The following figure illustrated such a mirrored scenario:

Figure 4 Remote heterogeneous wireless networks.

3 Load balancing model and Wardrop equilibrium

The model considered in this paper is an extension of a well-known model for selfish routing, where an infinite population of agents carries an infinitesimal amount of load each [5].

Let \( V \) denote the set of nodes connected to the set \( T \) of technologies. Each technology is characterized by a continuous, non-decreasing cost function (or originally mentioned, for instance in [5], latency functions) \( l_i: [0,1] \rightarrow \mathbb{R}^+ \). Moreover, let \( C = \{1, \ldots, c \} \) be the set of \( c \) commodities (i.e., traffic flows) to be deployed onto the heterogeneous network: each commodity \( i \) is characterized by a source node \( s_i \), a destination node \( d_i \), and a bandwidth (or transmission rate) \( r_i, i \in C \). Without loss of generality, for the sake of simplicity, in the following, commodity \( i \) is identified by \( s_i \) and \( d_i \) — i.e., two commodities with the same couple \( s_i, d_i \) do not exist).

An instance of the load balancing game is \( \Gamma = \{ V, T, (l_i)_{t \in T, i \in C} \} \). For \( t \in T \), let \( f_i^t \) be the bandwidth of commodity \( i \) assigned to technology \( t \). A population or flow vector \( (f_i^t)_{t \in T, i \in C} \) is feasible if, for all \( i \in C \), \( \sum_{t \in T} f_i^t = r_i \). Let \( f_i = \sum_{t \in C} f_i^t \) denote the load of technology \( t \in T \). Then, the latency of a resource \( t \) is function of \( f_i \), i.e., \( l_i(f_i) \).

Hereafter, the load balancing problem will be formulated as the problem of determining the strategies which will lead the flow vector to reach the Wardrop equilibrium. To this extent, the definition of agent is required. As defined, for instance, in [6], each agent is an infinitesimal portion of a specified commodity, whose objective is to minimize the cost sustained to reach its destination by a proper flow assignment. Practically speaking, a single packet of the flow could be approximately considered as an agent: in fact, even if the number of packets is finite, if the flow rates are sufficiently high, the population acceptably approximates the infinite population constraint required by the Wardrop theory ([5]).

In the Wardrop theory (see [5]), stable flow assignments are the ones in which no agent (i.e., no “small” portion of a commodity directed from a source to a destination) can improve its situation by changing its strategy (the set of used technologies along with the assigned bandwidth) unilaterally. This fact can be obtained if all agents reach the Wardrop Equilibrium.

**Definition 1** [5]: A feasible flow vector \( (f_i^t)_{t \in T, i \in C} \) is at a Wardrop equilibrium for the instance \( \Gamma \), if, for every commodity \( i \in C \) and every technologies \( t \) and \( t' \) in \( T \) with \( f_i^t > 0 \), it holds that \( l_i(f_i^t) \leq l_i(f_i^{t'}) \).

Practically speaking, at the Wardrop equilibrium all the technologies have the same latency function, leading to a fair exploitation of the resources and, as a consequence, better performances. To determine the cost of each agent, it is crucial to properly define the latency function \( l_i \), as defined before. An important parameter for a latency function, which is of interest for the convergence of the algorithm, is the so-called relative slope. Formally, as explained in [5], a differentiable latency function \( l(x) \) has relative slope \( d \) at \( x \) if \( l'(x) \leq d l(x) / x \). A latency function has relative slope \( d \) if it has relative slope \( d \) over the entire range \([0,1] \). Practically speaking, to reach the equilibrium, the agents could need to change the currently used technology (re-routing): the re-routing could not guarantee convergence to the Wardrop equilibrium if the latency functions make arbitrarily large leaps due to minor shifts of the flow. To restrict the number of agents migrating simultaneously, some information about the behavior of the latency functions must be known (this information is represented by the relative slope).

4 The load balancing algorithm

[5] defines an algorithm to dynamically learn a Wardrop equilibrium efficiently and in a distribute fashion. The algorithm is principally based on the concept of balancing among exploitation and exploration. This balancing guarantees the convergence of the algorithm to the Wardrop equilibrium. This approach is not new, and it is mainly used in several learning techniques (see for instance the Q-Learning approach [7], [9]) to let the algorithm converge eventually to the optimal solution and provide good solutions in the transitory phase. The exploitation policy, for a learning algorithm, simply consists in using the best action computed by the algorithm so far. The exploration policy, oppositely, aims at trying new, unexplored, actions in order to estimate its goodness: eventually, all the actions will be tested at least once and the algorithm converges to the optimum. The main challenge in defining specific exploitation-exploration policies is represented by (i) which action to take and evaluate in a determined time period (what is the difference with the current best action), and (ii) which is the interval of exploration and its related probability of exploration.

The following round-based approach is reported here and consists of an application of the approach proposed in [5], to the scenario considered in this paper. In every round
(each round starts every $T_{CONTROL}$ seconds), an agent is activated with constant probability $\lambda$. Let us consider an agent in commodity $i \in C$ currently utilizing technology $t$; the algorithm performs the following two steps:

1. **Sampling**: with probability $(1-\beta)$ perform step 1(a) and with probability $\beta$ perform step 1(b).
   (a) **Proportional sampling**: sample technology $u \in T$ with probability $f_u/r_i$.
   (b) **Uniform sampling**: sample technology $u \in T$ with probability $1/|T|$.

2. **Migration**: if $l_u < l_t$, migrate to technology $u$ with probability $(l_t - l_u)/[d(l_t + \alpha)]$.

The parameter $\alpha \geq 0$ can be chosen arbitrarily (introduced to avoid possible division by zero) and the parameter $\beta$ determines the balance among proportional and uniform sampling.

Once defined the migration policy, it is needed to specify the amount of flow that should be shifted between any pair of technologies within one round. For an instance $\Gamma$, let $d \geq 1$ be an upper bound on the relative slope of the latency functions. For every commodity $i \in C$ and every technology $u \in T$, with $l_u \leq l_t$, the $(\alpha, \beta)$-exploration-replication policy migrates a fraction of:

$$\mu_{iu} = \lambda \frac{1}{d} \frac{l_t - l_u}{l_t + \alpha} (1 - \beta) \frac{f_t}{r_i} + \beta \frac{1}{|T|}$$

agents from technology $t$ to technology $u$. [5] proves that such exploration-replication policy, applied to a multi-hop network with paths instead of single-hop technologies and in a centralized environment, converges, towards a Wardrop equilibrium. Some simulation results in ad-hoc networks are given in [12]. We evaluate convergence results and performances in the considered decentralized environment.

The most important factor of the above-described algorithm is the latency function, which intrinsically contains the information about the behavior, of each link, when a particular joint strategy (for all commodities) is chosen. This paper considers a latency function able to provide useful metric of the status of the network, when a determined joint action is applied. In particular, the packet loss probability of a technology, which represents one of the main problems of wireless networks, is chosen as the input for the latency function. Such metric is important because includes, in an aggregate value, the effects of several causes, such as the congestion of one link due to overloading, the collisions among different transmitters under the same transmission range, the noise, the interference with external transmitters, etc. Moreover, it is a technology-independent metric, in the sense that expresses in an abstract way (i.e., regardless of all the specific characteristics of a network, in terms of physical and protocol aspects) the current status of all the available technologies.

The packet loss probability $q(f_t)$ of a technology $t$, estimated in a pre-fixed time period $(T_{CONTROL})$ between the two rounds $k-1$ and $k$, is defined as the number of packets lost during the transmission onto technology $t$, over the total number of transmitted packet to $t$ in the considered time period. This paper proposes to associate technology $t$ with an exponential latency function such as:

$$l_t(f_t) = e^{\lambda q_t(f_t)}.$$ 

The use of the exponential associates a high cost also to relatively modest values of the packet loss probability. In this way the difference $l_t - l_t$ in point 2 of the presented algorithm is sensible also for modest values of the loss, resulting in a higher migration probability and, thus, in a more rapid convergence to good policies.

Since the packet loss probability is an instantaneous value (i.e., computed over the last time interval $T_{CONTROL}$), it could not represent the real condition of a network technology, but only a spike value. In common practice [7], the packet loss probability is used as an averaged value, over the time, of the instantaneous packet loss, computed according to the following update rule:

$$q_{t}^{OLD} = (1 - \gamma)q_{t}^{OLD} + \gamma q_{t}^{NEW},$$

where $\gamma \in [0,1]$ is the so-called learning rate parameter, which favor new estimates (large values) or the old ones (small values).

## 5 Simulations

The proposed approach was implemented in MATLAB, by extending an event-based framework available at [8]. The simulated wireless technologies are characterized by the CSMA/CA medium access protocol, with different physical transmission bandwidth. It is assumed that each device is able to easily measure the packet loss of each technology (for instance by contacting the related Access Point at each round). The proposed algorithm was compared with two single-path algorithms (i.e., algorithms which does not allow a flow to be transmitted over multiple paths) which choose (i) a unique technology regardless the current packet loss, (ii) the technology with the smallest packet loss. The algorithm is executed, in each node, every $T_{CONTROL}$ seconds, with the new collected packet loss information. Table 1 shows the algorithms parameters used in this simulation:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>0.001</td>
</tr>
<tr>
<td>$d$</td>
<td>1</td>
</tr>
<tr>
<td>$\beta$</td>
<td>0.7</td>
</tr>
<tr>
<td>$T_{CONTROL}$</td>
<td>0.5</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>0.5</td>
</tr>
<tr>
<td>$K$</td>
<td>25</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 1 Simulation parameters.

## 5.1 Load balancing scenario

The first scenario shows the capability of the proposed algorithm to make use of the availability of a set of distinct technologies and to satisfy the throughput requirement of the applications. Three users and three wireless technologies characterize this scenario. Each technology
has a physical bandwidth of 11Mbps and the resource access is governed by the CSMA/CA protocol. Each user transmits a unidirectional flow with Constant Bit-Rate (CBR) of 7Mbps. The other simulation parameters are represented in Table 1.

Figure 5 Total traffic received at the destinations with the Wardrop load balancing.

Figure 5 shows the total traffic received at the destinations of the three flows. As we can see, after a short transitory phase (about 5s), the algorithm is capable to satisfy the throughput requirement of the users applications.

Figure 6 Load balancing splitting.

Figure 6 allows the reader to understand the usage of the three network resources. At $t=0$ (initial state) the traffic is completely sent to the first technology and the traffic received is saturated at about 8Mbps (the actual delivery rate of a saturated technology of 11Mbps of nominal rate: such difference is given by the protocols overhead, e.g. CSMA/CA). The feedback provided by the network (i.e., the estimation of the packet loss probability of each technology) allows to smoothly switch part of the traffic over the two other unloaded technologies. At about $t=5s$, all the available technologies are fairly exploited, producing a high throughput (as shown in Figure 5). Finally, Figure 7 shows the throughput comparison of the proposed algorithm with the approach which exploits a single technology (Single technology) and the approach which chooses, at each time step $k$, the technology with the least packet loss probability (Least Loss Selection). The Wardrop load balancing is capable to satisfy the throughput requirement. The Least Loss Selection algorithm produces better results than the Single technology approach but fails in achieving high performances, due to the continuous oscillations between the two technologies.

Figure 7 Load bal. scenario: throughput comparison.

5.2 Response to network feedback: total handover

The previous scenario showed the load balancing capability of the proposed algorithm, in a quite static scenario. The following simulation aims at proving that, thanks to the feedback provided by the network, the Wardrop load balancing is capable of performing a total handover among technologies in a small amount of time. One user and two wireless technologies characterize this scenario, each with 11Mbps of nominal bandwidth. The flow of the user consists of a CBR flow with bit-rate of 7Mbps. The other simulation parameters are listed in Table 1. Initially, the traffic is completely sent over the first technology (since no loss is experienced over the first technology). At $t=12s$, the packet loss probability of the first technology increases to about 0.5 due to channel noise. The high packet loss probability makes the algorithm decide to switch the traffic to the other available technology (which is experiencing no channel noise).

Figure 8 Handover scenario: throughput comparison.

Thanks to this rapid handover, the throughput (shown in Figure 8) rapidly (about 4s) converges to its maximum and the user throughput requirement is satisfied.

5.3 Performances over the number of devices

This last scenario is characterized by two wireless technologies, each with 11Mbps of nominal bandwidth. The number of devices is increasing from 10 to 50. Each device starts a CBR flow with an application bandwidth of 300kbps. The other simulation parameters are listed in Table 1 (in particular $T_{\text{CONTROL}} = 0.5s$). Figure 9 shows the throughput (in the steady state) over the number of devices of the proposed load balancing algorithm and the single technology approach. The single technology approach saturates at 30 devices whereas the proposed algorithm
continues linearly (the saturation point is at about 16Mbps), taking advantage of both the technologies.

The most important results are shown in Figure 10, which reports the average rise time (i.e., the period to reach, the first time, the throughput steady state value) of the proposed load balancing algorithm. The rise time values are low (the peak is 3.4s at 50 devices) compared to the introduced advantages, demonstrating the applicability of the proposed algorithm in real heterogeneous scenarios with multiple users.

![Figure 9](image9.png) **Figure 9** Throughput over the number of devices.

![Figure 10](image10.png) **Figure 10** Rise time over the number of devices.

6 Conclusions

This paper presented a load balancing algorithm based on the concept of Wardrop equilibrium, extended to cope with the issues of wireless heterogeneous networks (such as Wi-Fi, WiMAX, etc.). In particular, (i) a specific metric considering the packet loss probability of each technology is introduced to deal with the peculiarities of wireless networks, (ii) the Wardrop-based algorithm guarantees convergence to stable policies. Simulations show higher performances in comparison to the approaches, which choose (i) a unique technology and (ii) the technology with the smallest packet loss. Moreover, the convergence time is reasonable, demonstrating its applicability in real scenarios.

This work suggests several interesting future researches. A possible extension could be to dynamically choose the exploration parameter $\beta$, in function of the status of the technologies (e.g., by increasing the exploration in case a drastic change occurs in the network status). Moreover, the packet loss metric could be integrated with other metrics (e.g., power consumption, delay, etc.), within a multi-objective optimization framework. Finally, flows importance could be taken into account, to favour high priority flows in the resource assignment process.

7 Acknowledgements

The research leading to these results has received funding from the EU project FI-WARE, under Grant Agreement n° 285248 and the Italian project PLATINO, under Grant Agreement n° PON01_01007.

8 References


