An Evaluation Criterion for Adaptive Neighbor Selection in Heterogeneous Peer-to-Peer Networks

Marco Picone, Michele Amoretti, and Francesco Zanichelli

Department of Information Engineering, Univ. of Parma, Italy
picone@ce.unipr.it
http://dsg.ce.unipr.it

Abstract. The peer-to-peer paradigm potentially enables low-cost and highly scalable distributed systems where user nodes have at the same time the roles of consumer and provider of resources. Nowadays, the computational power and storage capacity of mobile devices has notably increased, for which their inclusion in the physical network that supports P2P overlays can be realistically considered. For those applications where the quality of service is a fundamental constraint, such as live streaming, it is important to have overlay network level strategies to dynamically reconfigure the active connections among peers. In this paper we propose a simple but effective strategy for the evaluation of neighbors, taking into account their capabilities, to support the dynamic selection of resource providers. We apply the formal framework to P2P Internet Live Streaming, simulating three realistic scenarios and discussing the results.

1 Introduction

Peer-to-Peer (P2P) applications have recently emerged as an effective solution for large-scale content distribution over the Internet, from the early Napster (1999) to today's widespread fully decentralized delivery of multimedia streams, without relying on the traditional client-server paradigm.

In general, P2P approaches potentially enable low-cost and highly scalable distributed systems where user nodes have at the same time the consumer and provider roles. Sharing their own resources, peers contribute to the achievement of a global task which needs only a limited (if any) infrastructure since it exploits the large pool of user resources. *P2P Internet Streaming* is becoming one of most popular services on the global network and by means of application-level multicasting techniques achieves large-scale distribution of massive amounts of data with strong temporal constraints without compromising the provided quality of service [5,7,8]. These applications are mostly targeted to a set of wired nodes which are considered heterogeneous only with respect to the connection type and speed.

Recent years have also seen the relentless market success of a plethora of mobile devices (PDAs, smartphones, MIDs, PMPs, netbooks, . . .), whose ever

increasing capabilities make them attractive to a growing number of network applications in *business* and *infotainment* domains to be fully experienced in mobility. Although the computing, storage and communication resources available on many modern mobile devices are often not far from those available on common PCs, the specific issues of reduced connection stability (because of being wireless and moving across different access networks) and limited autonomy (because of being battery-powered) should be addressed in any distributed application which includes support for mobile nodes.

This work presents an adaptive peer selection strategy for heterogeneous peer-to-peer networks, with particular reference to Internet live streaming applications. The prioritization strategy supports extremely heterogeneous P2P networks where nodes exhibit very diverse nature and performance. Given that neighbor selection, *i.e.* one the critical functions of any P2P system, is generally based only upon the availability of required resources and the evaluation of the bandwidth capabilities of prospective partners, a more general selection criterion is required to cope with node heterogeneity. The adaptive selection strategy in our approach compares nodes by taking into account a certain number of their features (as required by the application domain) to the purpose of increasing the experienced quality of service by performing uniform evaluation of highly heterogenous peers.

The paper is organized as follows. Section 2 outlines a brief review of the main challenges to be faced when designing P2P applications supporting mobile devices. In section 3 some related work is discussed. Section 4 provides a formal characterization of the evaluation criterion for peer selection and introduces a set of node features appropriate for P2P live streaming applications.

In section 5 the proposed selection strategy is preliminarily evaluated by means of simulation on a number of heterogenous P2P live streaming scenarios.

2   Design Challenges

So far, P2P applications have been mostly designed and implemented considering personal computers as hosts and resource providers, rather than mobile devices. Nowadays, the computational power and storage capacity of mobile devices has notably increased, for which their inclusion in a physical network that supports a P2P overlay can be realistically considered. In this section we recall some general considerations to better define the issues that arise when mobile devices participate in a distributed system [6].

Different interfaces for data connectivity. In particular, mobile phones may have different kinds of data connections (*e.g.* CDMA/GPRS/3G) or in addition, as in the case of most modern devices, they can access WiFi networks. Applications like P2P media streaming are very sensitive to delays and loss of segments. For all these reasons, the different nature of connections and the possibility of sudden switches from an access network to another, are factors to take into account in order to dynamically adapt the overlay network to the characteristics of individual nodes.

Intrinsic mobility of devices. Physical mobility may involve changes in access speed to the network, but can also cause prolonged or momentary disconnections of peers, more frequently with respect to peer hosted by personal computers. Redundancy of resources and quick discovery of new providers is the general solution to this problem, but a purely reactive approach is not sufficient for applications like P2P multimedia streaming, for which proactive strategies should pursued instead (*e.g.* each peer should have a list of suitable neighbors, in the sense that if a new segment provider is selected when the current one disconnects from the network or offers poor performance, packet losses are minimized).

Limited resources. As mentioned above, with respect to personal computers, mobile devices are characterized by limited battery autonomy low memory and reduced computational power. This aspect is very important and requires several optimizations for the software to be run on the device, in order to guarantee a good quality service. Compared to the previous issue, this one is fortunately becoming less and less relevant, since an increasing number of mobile devices with very high autonomy and generous computational and storage capacities is entering the market. These new mobile devices will allow the developers to create richer applications, improving the experience for the user.

3   Related Work

Dynamic neighbor evaluation and selection is used in several P2P systems described in the literature.

In KaZaA [3] and Gnutella2 [2], the most powerful nodes (*e.g.* those with higher bandwidth) are set as super-nodes to form the backbone of the P2P overlay network, which manages most of the messaging load for resource discovery. In BitTorrent[1] systems, central servers store information about trackers, *i.e.* agents that are responsible for helping peers find each other. The BitTorrent protocol focuses on high data transfer speed rather than on search capabilities. When joining the torrent, the peer asks to the tracker a list of IP address of peers to build its initial peer set, *i.e.* the list of other peers it knows about. A peer can only send data to a subset of its peer set, called active peer set. The "choke algorithm" determines the peers being part of the active peer set. Each downloader reports to all its peers what pieces it has, thus each peer knows the distribution of the pieces for each peer in its peer set. The piece selection strategy used in BitTorrent is based on rarest first strategy.

Kwong et al. [9] propose a simple but interesting protocol for building heterogeneous unstructured P2P networks. The basic idea is to take into account node capacity during the joining and rebuilding process. In a first step, incoming new peers select as neighbors those peers that are more suitable in terms of capacity and connectivity, with the purpose of achieving good load-balancing. The rebuilding process describes how the nodes act to re-establish an efficient topology when some of their links are broken. To the same purpose, *i.e.* dynamically re-shaping the overlay network topology, interesting strategies based on genetic algorithms have ben proposed [10,11].

In applications like P2P Internet live streaming, where the quality of service (QoS) plays a very important role, neighbor selection must be carefully performed. The basic solution is random neighbor selection from the list received by the bootstrap node of the system or by a decentralized mechanism. A better solution is adopted by other protocols like CoolStreaming[5], where nodes periodically exchange information about their status, *e.g.* about the stream segments that they own and they can share, and about their actual bandwidth. These information are used by the provider selection strategy during the start-up process of each single node, and for managing changes of provider during the life of the peer.

To create a P2P streaming protocol that can be applied to a network of heterogeneous hosts, including mobile devices, it is important to introduce a peer selection criterion that dynamically evaluates the QoS of the neighbors, in order to personalize the behavior of each peer. A work of Nemati et al. [7] tries to analyze and model a network with mobile and heterogeneous devices in terms of mobility and QoS. The protocol defines two kinds of mobile peers: indirectly mobile peers whose hosts are mobile, and directly mobile peers that are realized as mobile agents. A mobile peer can manipulate a multimedia content in a host. QoS supported by the peer depends on the host. Movements of mobile peers are modeled in terms of changes of QoS, which is described and characterized by bandwidth, delay and packet-loss-ratio under overlay level and at the overlay by frame-rate, resolution, number of colors, quality of sound, etc. With this model the peers of the network can analyze the parameters of other nodes to select the best provider, trying to maximize the obtained QoS in a dynamic way.

4 Formal Statement of the Evaluation Criterion

We consider a network with N peers as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges. Each peer has a number of resources, whose domain is defined as $\mathcal{R}$. Moreover, each peer can distinguish principal resources, in a set $\mathcal{R}_p$, from secondary resources, in a set $\mathcal{R}_s$, such that $\mathcal{R} = \mathcal{R}_p \cup \mathcal{R}_s$.

The status of both main and secondary resources should be taken into account for a peer that has to be evaluated by another peer. But in a heterogeneous network some kinds of peer could be lacking one or more kinds of resources, with respect to other richer peers. For this reason we introduce a unique parameter to characterize any kind of peer, *i.e.* the *fitness*. In other words, we state that for each peer in V $\exists f_i > 0$ defined as:

$$f_i = f(g_i, h_i) = \frac{g_i(\mathbf{r}_s)}{K_0 + h_i(\mathbf{r}_m)} \quad K_0 > 0 \quad \mathbf{r}_s \in \mathcal{R}_s^{m_i} \quad \mathbf{r}_m \in \mathcal{R}_p^{n_i} \tag{1}$$

where

$$g : \mathcal{R}_s^m \to \mathbb{R} \quad m \in \mathbb{N} \tag{2}$$

and

$$h : \mathcal{R}_m^n \to \mathbb{R} \quad n \in \mathbb{N} \tag{3}$$

As illustrated in figure 1, the fitness is a family of curves that depend on the value of functions h and g. For a given peer i, with n_i principal resources and m_i secondary resources, the fitness value and h_i are inversely proportional, while the fitness value and g_i are directly proportional. The curve progress of functions g, h, as well as the value of K_0 must be defined based on the application, always avoiding the following situation:

$$f_1(h_1, g_1) > f_2(h_2, g_2) \quad h_1 > h_2 \tag{4}$$

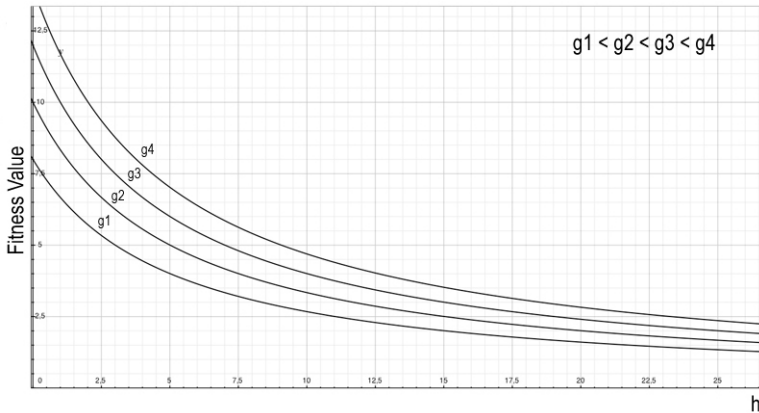in which the node with lowest h has the highest fitness value.

Fig. 1. Fitness curve progress with respect to h and g

This evaluation criterion can be used in a wide range of P2P applications. For example we consider a P2P Internet Live Streaming application based on the data-driven approach, where the stream is distributed depending on the availability or need of data chunks, unlike traditional Internet streaming systems (*e.g.* those based on the source-driven paradigm) that perform explicit search of the optimal streaming paths from the source(s) to the sinks. The fitness-based approach introduces adaptiveness in the data-driven system, to maintain a high degree of efficiency even in presence of heterogeneous nodes with different availability profiles (*e.g.* mobile nodes versus stable nodes). Each node is periodically evaluated according to the status of its resources, *i.e.* its dynamic features (residual battery, type of network connection, available bandwidth, on-line permanence time), with a fitness value being computed. By comparing the fitness of its neighbors (*i.e.* known peers), each peer can select the locally optimal multimedia stream providers. This allow to reduce the impact on the system of least performing nodes (*e.g.* nodes with reduced bandwidth and computational capacity), and to improve the overall quality of service.

The fitness value of a single node is computed considering the following parameters (which represent resources characterizing every node):

- actual upload bandwidth (U): the nominal transmission rate of a node, over the number of neighbors being served by that node
- battery percentage (B): in case of PC node, its value is always 100; for a mobile node, the value decreases depending on node usage
- on-line permanence time (T_o): its value defines how long the peer has been connected to the network

Among these resources, we considered U as principal, while B and T_o are secondary. Since all nodes are characterized by the same number of resources (*i.e.* $m_i = m$ and $n_i = n$ for each $i = 1, .., N$), the h and g functions are the same for every node. Supposing a node i evaluates another node j, the h function applied to the latter is defined as

$$h(U_j) = \left(\frac{min\{D_i, \frac{U_j}{k_j+1}\}}{S} \right)^{-1} \tag{5}$$

where D_i is the download rate of node i, S is the stream rate of the multimedia source assumed to be constant, and

$$k_j = \sum_{l=1}^{N} e_{lj} \tag{6}$$

where $e_{lj} \in \{0,1\}$ has non-zero value if node l is being serviced by node j. In (5), the numerator represents the transmission rate that will be devoted to node i from node j, if selected. U_j is evaluated with respect to already active outgoing connections (which in general are less than the number of known nodes by each peer), plus the envisioned connection (for which neighbors are being evaluated by node i).

The g function applied to node j is defined as

$$g(B_j, T_{oj}) = K_1 \frac{B_j}{B_{Max}} + K_2 \frac{T_{oj}}{T_{oMax}} \quad K_1 > 0 \quad K_2 > 0 \tag{7}$$

Thus the fitness function applied to node j is

$$f_j = f(g_j, h_j) = \frac{g(B_j, T_{oj})}{K_0 + h_j(U_j)} \tag{8}$$

Fitness-based evaluation can be applied to the following phases of a generic data-driven protocol (see also figure 2):

1. *Neighbor list update*: to limit the negative effects of random disconnections and to monitor the quality of service of the system, each peer periodically updates its neighbor list; this is done by sending messages to all known nodes, which in turn respond with their updated resource values; if a contacted node does not respond before a given deadline, it is removed from the neighbor list, and replaced with another node.
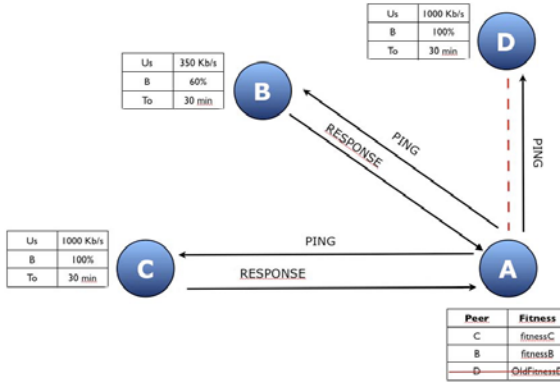
Fig. 2. Example of neighbor list update, performed by node A

2. *Provider selection*: once the peer has updated its neighbor list, it selects the
   neighbor with the higher fitness value to be the provider of the multimedia
   stream.

5   Simulation Results

The adaptive peer selection strategy applied to live streaming has been imple-
mented and preliminary evaluated using the Discrete Event Universal Simulator
(DEUS) [4], which is an open source tool that provides a simple Java API for
the implementation of nodes, events and processes, and a straightforward but
powerful XML schema to configure the experiments. DEUS aims to be one of
the reference tools in the field of complex system simulation, thanks to its high
flexibility and performance.

Three node types have been modelled with the parameters illustrated in
Table 1: PC with high performance in terms of computational capacity and
bandwidth, Mobile-WiFi nodes with limited resources but accessing the net-
work with a Wireless Lan connection, and 3G Mobile nodes using a connection
with lower bandwidth and subject to the variability of cellular networks.

For the behavioral modeling of mobile devices two types of specific events
have been considered:

Table 1. Parameters of the simulated nodes

Node type	Uplink	DownLink	Battery
Mobile3G-Node	50 Kbit/sec	400 Kbit/sec	[20,100]
MobileWiFi-Node	100 Kbit/sec	2 Mbit/sec	[20,100]
PC-Node	300 Kbit/sec	4 Mbit/sec	100%

– The transition from a 3G to a 2G connection, *i.e.* a connection with low performance that has a reduced uplink bandwidth of only 25Kbit/sec and a downlink bandwidth of 100 Kbit/sec.
– Battery consumption; each peer enters the network with a random fraction of battery capacity; this value (Q_B) is periodically decreased according to elapsed time and to the number of outgoing active connections:

$$Q'_B = f(Q_B, k_{out}) \qquad (9)$$

In general, we assumed that with a charge of 100%, a mobile peer can stay active for 1 hour and 30 minutes of playback. When the remaining battery falls below a certain threshold, the node leaves the system with a consequent reconfiguration of the peers which were directly connected to it.
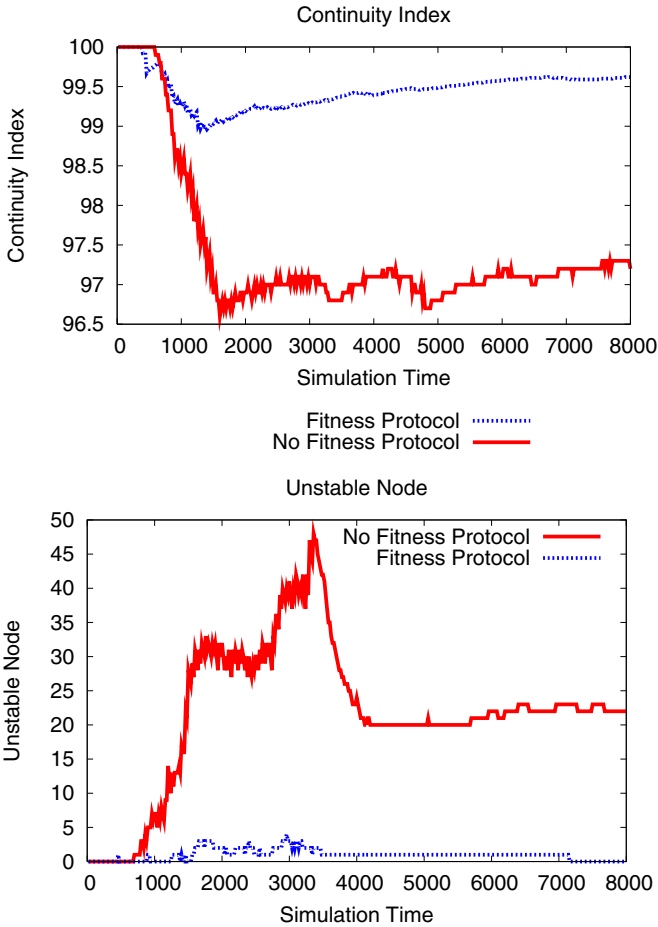

Fig. 3. Comparison of Continuity Index and Unstable Nodes in the first scenario

Initially we considered a simple data-driven protocol [8] in which each node has a single segment provider. We compared the peer selection strategy based on the evaluation of the fitness value associated to each node with a simple random selection of the supplier for the media stream.

The first analyzed scenario has a higher percentage of PC nodes and Mobile-WiFi and consequently a large amount of available resources. Applying the fitness strategy allows to select the most efficient suppliers, reduces the load on the source node and prevents the undesirable effect of having Mobile Wi-Fi and 3G nodes that become suppliers for subgroups of nodes. Obtained simulation results (figure 3) show a small percentage of links to nodes with poor performance, thus increasing the Continuity Index (CI), which is the percentage of video segments received before the playback deadline, and reducing the number of unstable nodes in the system (*i.e.* nodes with a CI under 90%).

The second scenario shows a higher number of Mobile3G compared to other types of nodes and implies a reduced availability of resources. In this situation, when no evaluation of the nodes is performed, a remarkable reduction of the CI can be noticed along with a significant increase of unstable nodes in the system as in figure 4. The reason is that selected nodes are inefficient suppliers
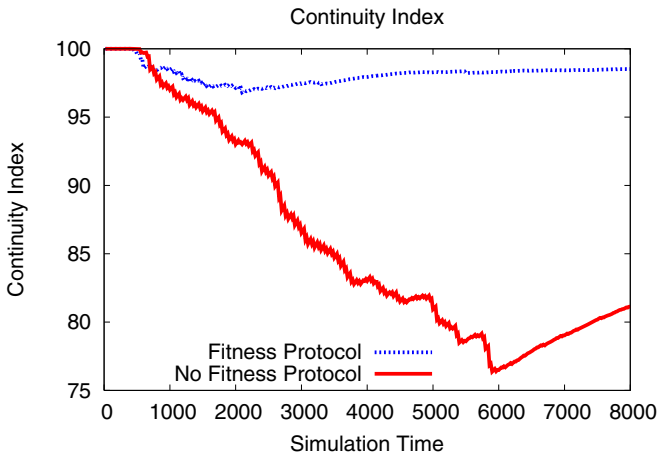

Fig. 4. Comparison of CI obtained with a higher number of 3G Nodes

Table 2. CoolStreaming simulations parameters

Name	Value
Total Nodes	712
Num. Of PC-Node	396
Num. Of MobileWiFi-Node	141
Num. Of Mobile3G-Node	179
Simulation MaxTime	110000 VT
Real MaxTime	1:30 h

for their subgroups of peers in the system. This choice involves a reduction of performance and a progressive deterioration of the system. Instead, the use of the fitness-based strategy reduces these negative effects, trying to exploit as much as possible the ability of the most efficient peers.

The increase of the CI without fitness criterion after 6000VT is a consequence of the disconnection of mobile nodes that in this scenario have to manage a lot of connections, which cause a fast decrease of the battery level. At the end of the simulation there are less mobile nodes compared with the beginning of the simulaton, for which PC nodes can manage the distribution of the media stream with better performance, in terms of continuity index.

Obviously the impact of mobile nodes is very important in a system with only one segment provider per peer, where the disconnection or the presence of low-performing intermediate providers readily affects the system's performance. More
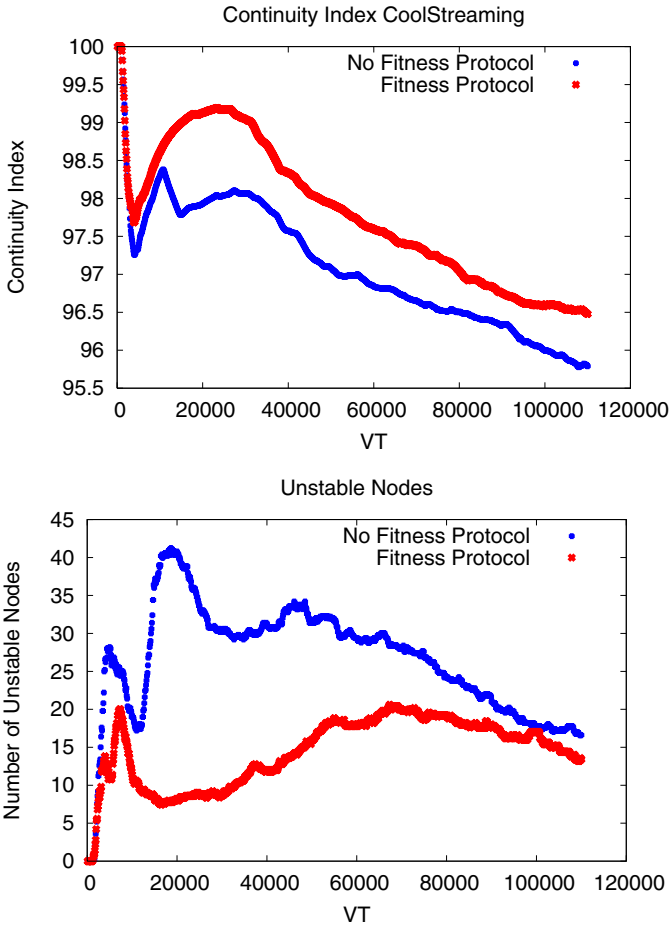

Fig. 5. Unstable Nodes and Continuity Index in CoolStreaming, with and without fitness

complex protocols, involving multiple suppliers per peer, are more robust against temporary loss of suppliers, because the responsibility of the flow is distributed among multiple nodes. For this reason, the second phase of analysis of the fitness criterion has been focused to its application to the popular multi-supplier protocol for P2P live streaming, *i.e.* CoolStreaming [5]. This protocol assigns a set of segment providers to each peer, with an evaluation strategy that analyzes the available bandwidth and resources of nodes with which the other peers come into contact. This value is used for the selection of the appropriate provider for the media-stream. For the reasons explained previously in a network that shows heterogeneity in terms of devices, a more detailed and specific strategy is necessary, for which we added the fitness-based strategy to the CoolStreaming protocol. Simulations have been carried out using DEUS, using the same types of nodes and events that have been used for the analysis of the simple data-driven protocol. The evaluation is based on the characteristics listed in Table 2.

In this second analysis the fitness strategy replaces only the bandwidth strategy of CoolStreaming, not the stream segments evaluation that is to much tied together with the protocol and can not be changed. The results show that also in case of a real system like CoolStreaming with a high heterogeneous network in term of kinds of devices, connections and behaviours can experience an increase of performance using the fitness selection strategy. Figure 5 shows that during the simulated time there is an increase of CI and a decrease in the number of unstable nodes. Near the end of the simulation, where the number of devices and the global battery level are reduced, there is still a small group of high performance nodes. For this reason there is a reduced range of good potential providers
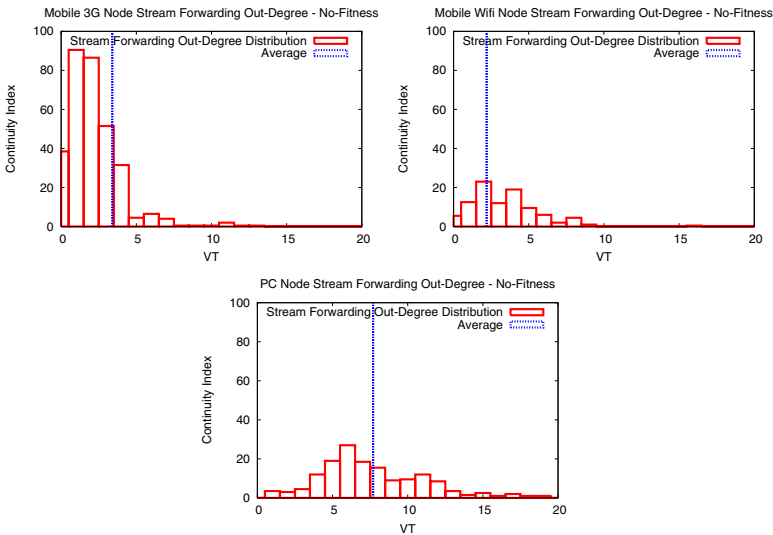

Fig. 6. Distribution of the number of outgoing active connections ($k_{o}ut$) for each peer, for the standard CoolStreaming protocol
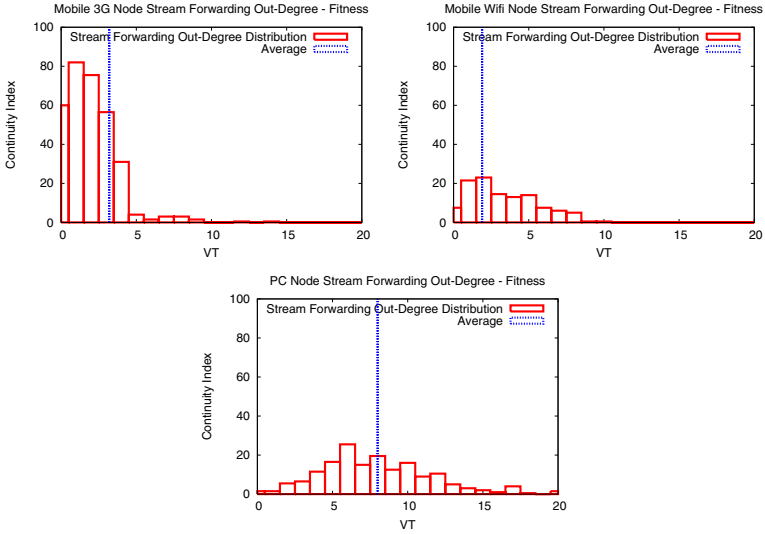
Fig. 7. Distribution of the number of outgoing active connections (k_out) for each peer, for CoolStreaming protocol enhanced with fitness-based strategy

and the difference of performance between the two approaches is less evident. In figures 7 and 6 we compare the stream forwarding out-degree distribution (at simulation's virtual time 50000) without and with fitness-based strategy for each kind of node. They show that with the adaptive strategy there is a little advantage and that nodes with higher capacities are used more than nodes with lower capacities.

6   Conclusions

In this work we have proposed an evaluation criterion for neighbor selection in distributed systems based on the peer-to-peer paradigm, where peer processes are hosted by heterogeneous devices, including mobile devices. Our approach is based on the concept of fitness, which allows to compare resource providers with different characteristics. We have shown that the general approach can be specialized to a particularly challenging problem, *i.e.* P2P Internet Live Streaming. Simulation results are encouraging, showing that the fitness-based strategy can be applied to existing protocols, with effective performance improvements.

As future work, we will use our strategy in conjunction with techniques and protocols that have been specifically defined for constrained devices. For example, in the field of P2P Internet Live Streaming we will consider layered video coding, and multiple description coding. Those techniques allow to adapt multimedia content to differently featured devices. Moreover, we will investigate the application of the fitness-based strategy to other P2P applications, such as data-intensive computing and multi-player online gaming.

References

1. BitTorrent official site, `http://www.bittorrent.org`
2. Gnutella official site, `http://www.gnutella.com`
3. KaZaA official site, `http://www.kazaa.com`
4. Amoretti, M., Agosti, M., Zanichelli, F.: DEUS: a Discrete Event Universal Simulator. In: Proc. of the 2nd ICST/ACM International Conference on Simulation Tools and Techniques (SIMUTools 2009), Roma, Italy (March 2009)
5. Keung, G.Y., Xie, X.Z.S., Li, B.: CoolStreaming: Design, Theory, and Pratice, pages. IEEE Transactions on Multimedia 9(8), 1661–1671 (2007)
6. Ahmed, T., Mushtaq, M.: P2P-based Mobile IPTV: Challenges and Opportunities. In: Proc. Computer Systems and Applications, AICCSA 2008. IEEE/ACS International Conference, pp. 975–980 (2008)
7. Takizawa, M., Nemati, A.G., Enokido, T.: A Multi-Source Streaming Model for Mobile Peer-to-Peer (P2P) Overlay Networks. In: Proc. The 28th International Conference on Distributed Computing Systems Workshops, pp. 18–23 (2008)
8. Fourmaux, O., Silverston, T.: Source vs Data-driven Approach for Live P2P Streaming. In: Proc. International Conference on Networking, Internation Conference on Systems and Intenation Conference on Mobile Communications and Learning Technologies, pp. 99–104 (2006)
9. Kwong, K.W., Tsang, D.H.K.: Building heterogeneous peer-to-peer networks: Protocol and analysis. IEEE-ACM Transactions on Networking 12(2) (April 2008)
10. Koo, S.G.M., Kannan, K., George Lee, C.S.: On neighbor-selection strategy in hybrid peer-to-peer networks. Future Generation Computer Systems 22, 732–741 (2006)
11. Abraham, A., Yue, B., Xian, C., Liu, H., Pant, M.: Multi-objective Peer-to-Peer Neighbor-Selection Strategy Using Genetic Algorithm. In: Aluru, S., Parashar, M., Badrinath, R., Prasanna, V.K. (eds.) HiPC 2007. LNCS, vol. 4873, pp. 443–451. Springer, Heidelberg (2007)