Network coding and evolutionary theory for performance enhancement in wireless cooperative clusters†

Leonardo Militano1*, Frank H. P. Fitzek2, Antonio Iera1 and Antonella Molinaro1

1DIMET Department, University Mediterranea of Reggio Calabria, Via Graziella, Località Feo di Vito, Reggio Calabria 89060, Italy
2Aalborg University, Denmark

SUMMARY

Cooperation over short-range wireless links among user devices, which download remote contents through cellular links, is a paradigm quickly gaining ground, given that it can answer several technological and design issues that next-generation wireless applications will raise. Among others, energy consumption, data rate and data transfer time are parameters that will likely benefit from this novel communication concept. This paper presents a WLAN-based cooperative scenario, conceived to support file-sharing services, in which a sub-set of cluster nodes access, through their cellular radio interface, portions of a file to be successively exchanged among all cluster members over wireless local area network (WLAN) links. Besides showing the beneficial effects of cooperation, this paper also focuses on the performance enhancement that can be achieved when using the network coding paradigm, whose deployment revealed to be very effective in the depicted scenario. The selection of the nodes acting as sources for file injection into the cluster is a critical aspect for the effectiveness of the whole system. Therefore, the potentialities offered by the evolutionary theory applied to the specific problem are investigated and an ad hoc conceived Genetic Algorithm (GA) designed. Either the service time (the time needed for all nodes to receive the complete file) or the energy consumption for the nodes is used as objective function, showing in both cases the fast convergence for the algorithm that makes it preferable to either exhaustive searches or random choices. Copyright © 2010 John Wiley & Sons, Ltd.

1. INTRODUCTION

Wireless cooperation is a communication paradigm which fostered fervent research activities in recent years, to the purpose of applying to different protocol layers cooperation principles such as multi-hop/relaying, cooperative antennas, cooperative diversity and cooperative coding [1]. Specifically, the present paper focuses on a kind of ‘inter-system’ and ‘multi-network’ cooperative communication paradigm [2], which perfectly fits the scenario of fourth-generation communications systems. The underlying concept of the introduced communication paradigm is to merge short-range technologies and conventional cellular networks into a novel communication scenario. Typically, cellular networks rely on a remote centralised infrastructure which offers low data rate services in wide areas, while short-range networks are self-organising and infrastructureless, meeting the requirements of nomadic users, and ensure high data rates. Based on a peer-to-peer communication among involved devices, complementarities and capabilities of cellular and short-range radio networks may concur to offer significant advantages. Arising benefits are shared among network carriers, service providers and customers. If increased capacity, improved quality of service (QoS) levels, energy enhancements and spectral efficiency are the basis for the interest in this novel paradigm, the success is also sustained by the large number of services and applications this synergic relationship

* Correspondence to: Leonardo Militano, DIMET Department, Mediterranea University of Reggio Calabria, Via Graziella, Località Feo di Vito, Reggio Calabria 89060, Italy. E-mail: leonardo.militano@unirc.it
†A previous version of this paper was presented in the 15th European Wireless Conference (EW 2009), Aalborg, Denmark.

Received 1 December 2009
Revised 29 April 2010
Accepted 3 May 2010
between cellular and short range technologies paves the way for.

In this paper a wireless cooperative file-sharing service is presented, where multiple nodes cooperate on wireless local area network (WLAN) short-range links while being also connected to cellular links. More specifically, the reference scenario is a fully connected WLAN-based cluster in which $N$ multi-mode terminals are randomly scattered within a square area $A$ to jointly access a file-sharing service. The devices will cooperatively exchange over the short-range link the relevant data they have received either over the cellular link or from neighbouring devices. The fully connectivity condition ensures that all the nodes can be provided with the file. This condition is given by the following inequality, as given in Reference [3]: $N \geq \frac{(10 \cdot A)}{(\pi \cdot R^2)}$, where $N$ is the required number of nodes in the network and $R$ is the transmission radius. For the present study, the equality value is assumed.

Some (potentially all) nodes, acting as information sources for the cluster, will receive unique shares of the file over their cellular link while short-range radio links will be exploited to exchange data with the neighbouring nodes. Source nodes will own a partial amount of data since each data portion will be injected into the cluster only once. The presence of a decision taker entity is envisaged, like the Cooperation Server in Figure 1, which will select source nodes in the cluster and base its decision exclusively on the number of nodes within the cluster. All involved nodes are required to be active part of the short-range data exchange to obtain the complete file. There are two parameters that benefit from this framework, namely the time needed for all nodes to receive the complete file, which will be named service time, and the energy consumed by nodes to receive the file.

The purpose of the research work reported in this paper is not simply to show the improvements obtained by cooperation, as this aspect has already been addressed by the authors in References [4] and [5], rather the intended contributions are listed in the following:

1. To analyse the performance enhancement offered by the introduction of the network coding paradigm on the cooperative short-range transmissions. The results will show how separate analyses are to be made when the service time or, alternatively, the energy consumption is the parameter to minimise.

2. To study the cluster-head selection problem for the proposed cooperative file-sharing framework. Cluster heads are the nodes that will have access over the cellular link shares of the file to be exchanged among cluster members; thus, they will behave as source nodes for the cluster.

In a previous contribution of the same authors [6], a genetic algorithm (GA) has been proposed for the cluster-head election problem and its convergence to an optimal, or alternatively, a satisfactory sub-optimal solution has been demonstrated. Compared to previous contributions, in this paper, a more accurate analysis on the GAs potentialities and parameter definition is presented; the assumption that the number of information sources to be elected was fixed and a priori known is removed. Moreover, attention that was earlier only devoted to the service time parameter is here also focused on the energy consumption.

The remainder of this paper is organised as follows. The next section introduces the proposed approach for cooperative file downloading with network coding. Subsequently, some related work is presented about the network coding techniques and the cluster-head election problem. Thereafter, attention to the parameter definition for the proposed GA is given. The performance results are presented for the proposed scenario in the section preceding conclusion section, while the final section concludes the paper.

2. THE COOPERATIVE FILE SHARING WITH NETWORK CODING APPROACH

Besides the great potential of cooperation among mobile devices in terms of lower service time and energy consumption [4, 5], further enhancement in the efficiency of communication among the mobile cooperating devices is expected to be given by network coding. In the reference scenario for cooperative file sharing with network coding, a sub-set of
cluster nodes will download on their cellular links different portions of a remote file to be injected into the cluster and exchanged among all its members. These source nodes will then start forwarding packets over the short-range links.

Every node owning some data, received either from a neighbour node or downloaded through its cellular link, will re-transmit a (possibly) new linear combination of the received/stored packets to its neighbours. The so encoded information will be used by the nodes in the cluster to reconstruct the original data after decoding the received packets. As in doing so the number of needed packet transmissions over the short-range link is reduced, this is expected to beneficially impact on both the energy consumption and the service time experienced by the nodes.

With reference to the scenario depicted in Figure 1, a key issue of utmost importance for the overall system performance is the correct choice of the sources in the cluster, the so called cluster heads, (i.e. those nodes that are in charge of downloading data portions over their cellular links). In taking this choice, it shall be considered that the time needed to spread data to all nodes is both influenced by the cellular link capabilities and the positioning of the nodes within the cluster. Moreover, the number of cluster heads and their positioning within the cluster both influence the overall short-range energy consumption.

In the proposed system in Figure 1, a decision taker entity, the Cooperation Server, will select the cluster heads based exclusively on the knowledge of the number of nodes in the cluster. The number of sources that minimises the service time (or alternatively the average energy consumption) is not unique and fixed; oppositely, there is no chance to take the ‘best choice’ a priori and generalise it to all possible scenarios. Therefore, one of the main research objectives in this paper is the design of an algorithm able to provide the ‘best set of nodes’ to be elected as cluster heads in a short time, both in the optimistic case in which at least the number of sources is known and in the most general case in which there is no a priori knowledge about such number.

Classic cluster head election procedures used in several contexts (e.g. in mobile ad hoc networks or in sensor networks) do not fit to the stated problem, due to its specific constraints: in the reference scenario, all nodes are part of one single cluster; no limiting condition exists in the number of cluster heads, which potentially can be equal to the whole set of nodes; cluster heads receive only a partial and unique content of the file; every node is destination of all data; the decision taker has no information about battery level, cellular throughput or position of the nodes (usual parameters exploited in the literature for cluster head election solutions). These features make any clustering algorithm conceived for sensor [7] and/or mobile ad hoc networks [8] not applicable to the reference scenario.

Evolutionary theory is expected to have all the desired features to successfully address the cluster head election problem. In the proposed approach, during the first part of the file sharing process, an ad hoc implemented GA will use small packets of useful information to test possible source configurations. In this case, dummy packets are not used, but useful data are exchanged since the beginning. During this early phase, the GA will learn from the tests to evolve towards the best solution.

The GA algorithm can be designed to work well for both service time and for energy consumption optimisation. The fitness value of every single solution will be either the time needed to spread a given amount of data across all the network nodes or the average per node energy consumption in delivering all data to every nodes. At the end of the evolution, the best solution will give the number and the combination of nodes to be used for the file sharing service (for the remaining part of the file). As it will be shown, the fast convergence property of the GA for the specific problem makes it preferable compared to a random choice. The solution the GA will converge to is often the optimal solution. Sometimes, a sub-optimal solution, which is also satisfactory, can be found in a shorter time.

Some general considerations hold as far as the optimisation problem is concerned. The results presented in the No free lunch theorems for search, by Wolpert and Macready in Reference [9], affect all fields of optimisation algorithms. This theorem states that an optimiser pays for superior performance achieved on solving some problems with inferior performance achieved on others. Furthermore, for a uniform distribution of possible problems, the gains and losses balance precisely, and all optimisers have identical average performance. As reported in Reference [10], the old-fashioned vision that natural evolution is an ‘optimised optimiser’ that produces outstanding results under all circumstances is not real and the No free lunch theorem is valid also in this case. Based on these results, it is not the intent to state in this paper that the evolutionary-based algorithm proposed in this paper is surely outperforming any other optimisation algorithm, but it is performing better than an exhaustive search or an alternative random selection.

3. RELATED WORKS

3.1. Network coding

First introduced by Ahlswede et al.[11], network coding is a method that increases the throughput and obtains max-
imum information flow, by moving away from traditional approaches to send data through a network. Different from traditional data routing, according to which forwarded data are not modified, network coding foresees that packets can be combined (encoded) on their path to destination to form new representations of the original data. Encoded information will be used by the receiver to reconstruct the original data after decoding the received packets. A quick and simple implementation of network coding uses an eXclusive OR (XOR) operation to code and decode packets; notwithstanding, other solutions, much more efficient and powerful such as random linear network coding (RLNC), exist. According to this latter approach, packets received by a node are linear combinations of some original information packets. The node will then re-transmit a (possibly) new linear combination of the received/stored packets. Decoding at any node to obtain the data requires the correct reception of a sufficient number of independent combinations. A theoretical model for RLNC is available from References [12] and [13]. One way to check whether a packet is innovative or not is through Gaussian Elimination [14]. The additional energy cost caused by the coding/decoding step has been analysed in depth in Reference [15], for a real implementation of WLAN-based cooperative data exchange with network coding on mobile devices. The power level for coding operations has been measured and compared to the transmission, reception and idle time power levels for the WLAN link. It has been shown that in such a scenario the network coding itself has nearly no impact on the overall energy consumption compared to the wireless part. Therefore, without loss of generality, the energy cost for coding operations will be considered negligible, in the remaining sections of this paper.

Wireless cooperative distribution is a nice example of how network coding can be used to utilise bandwidth more efficiently. Studies have already been conducted in the direction of a first implementation and performance analysis of network coding in a cooperative scenario and have been presented in Reference [15]. Nevertheless, still a thorough analysis is needed to understand to what extent the use of network coding in wireless cooperative networks has a positive impact on the performance. In the scenario introduced in Section ‘The Cooperative File Sharing with Network Coding Approach’, every node owning some data checks whether any of its neighbours is interested in receiving that (i.e. if the coded packet is innovative for the neighbour). If this is the case, then the node generates and transmits to its neighbourhood a linear combination of all these file chunks. The performance improvement reached by using the Selective-Forwarding with network coding strategy instead of the Blind-Forwarding with network coding strategy is very high and, as shown in Reference [16], the service time parameter is reduced to one-third of the value. This relevant improvement justifies the small additional signalling costs and the additional implementation complexity required to support the approach. In fact, in a real implementation it is assumed that each node maintains a table with complete knowledge about the list of chunks at each of its neighbours. In order to build these tables and maintain them updated, some signalling exchange among nodes is foreseen. It is out of the scope of this paper to show a real implementation of the signalling process and to analyse how this impacts the final performance. Thanks to the reduced amount of information to be exchanged among nodes, the introduced overhead is assumed to be negligible with respect to the total energy consumption and service time values. A similar scenario has been studied in Reference [17], which presents the effects of a limited upload capability of the nodes on a network coding-based file sharing service.

Different from the approach followed in this paper, the two papers in References [16] and [17] consider a network coding implementation in a ‘non-cooperative’ scenario, for networks where multiple copies of the data are present and each source gets the full range of data portions of the file to be shared. Given that in the present reference scenario only a single copy of each data portion is present in the network, (i.e. any source node handles unique packets), the network coding behaviour and related performance are different, as shown in Section ‘Performance Evaluation’.

3.2. Genetic algorithms

The natural inspiration behind the evolutionary theory is that given a population of individuals, a natural selection process will take place with the survival of the fittest; thereby, the fitness of the population will constantly grow. This feature is the key for the application of evolutionary techniques and more specifically of GAs to optimisation problems.

Given an objective function to be maximised and a random set of candidate solutions from the domain of the objective function, a GA can evolve according to the fitness-based
4. THE GA PARAMETER SETTING

Before proceeding with the description of the results obtained during the simulation campaigns, it is worth briefly explaining the procedure for the designed GA parameter setting.

Big efforts have been put in research activities aiming at finding a general-purpose set of parameters, valid for numerous GA applications. Examples, which are obtained experimentally and applicable to numeric test problems, can be found in References [27, 28] and [29]. Tuning one parameter at a time is a common choice, although this may cause sub-optimal settings, since the interaction among parameters is actually very complex [30]. On the other hand, simultaneously tuning multiple parameters leads to enormous (and very time consuming) amount of experiments, and, most importantly, it does not guarantee that the choice is actually the optimal one. Also in case of parameter setting by analogy [30] (i.e. the use of parameter settings already successful for ‘similar’ problems), it is not sure that the similarity between problems implies that the optimal set of GA parameters is also similar. Last possibility is to perform theoretical analysis, which unfortunately is usually very complex and only applicable after significant simplifications.

The approach followed in this present paper consists in tuning the parameters one by one, basing the choices on experimental results. It is worth mentioning that also the idea of using a parameter control solution, like in References [30] and [31], has been considered, according to which parameters are dynamically updated during the GA running phase. Nevertheless, a marked performance deterioration has been noticed in this case, which pushed towards the adoption of a simpler tuning solution without parameter control.

The parameters on which the attention has been focused are the population size \( \mu \), the crossover probability \( p_c \) and the mutation probability \( p_m \). For these parameters different values have been proposed in the literature: \( p_m = 0.001 \), \( p_c = 0.6 \), \( 50 \leq \mu \leq 100 \) in Reference [27]; \( p_m \in [0.005, 0.01] \), \( p_c \in [0.75, 0.95] \), \( 20 \leq \mu \leq 30 \) in Reference [29] and \( p_m = 0.01 \), \( p_c = 0.95 \), \( \mu = 30 \) in Reference [28].

4.1. Population size \( \mu \)

The value of \( \mu \) represents the number of tests the GA will perform in the first generation. At the first generation, \( \mu \) initial chromosomes shall be selected. Different values are tested, always under the constraint that \( \mu \) is a multiple of the
number of nodes $N$, and the initial $\mu$ chromosomes are chosen randomly among all the possible combinations, under the condition to have an equal number of chromosomes for each source cardinality between 1 and $N$. Under this condition, equal ‘chances’ are given, during the evolution, to all possible numbers of source nodes. In Figures 2 and 3 (plots on the left), the GA performance in terms of service time units ($T$) for different values of $\mu$ is plotted. The goodness of the solution is assessed according to the number of tests needed before convergence to the optimal values (obtained through an exhaustive search). The test-case is a 12-node network, in which six nodes are served by high-speed downlink packet access (HSDPA) cellular links and six nodes by general packet radio service (GPRS) links. Nodes are randomly distributed across the network. As a result, $N$ is the best value for $\mu$.

4.2. Crossover rate $p_c$

At each generation, a random number, uniformly distributed between 0 and 1, is extracted; if it is smaller than $p_c$, then two parents are combined to produce two children, otherwise nothing changes. A one-point crossover solution has been used, which means that a random number $r$ between 1 and $L - 1$ is extracted, with $L$ the code length (the number of nodes in this case) and the two parents will exchange the tales from $r$ on. Different solutions for $p_c$ have been tested. Figures 2 and 3 (plots in the middle) show the GA performance for the different values of $p_c$ in the range $[0.6, 0.95]$. Also in this case, the goodness of the solution is assessed in terms of service time units ($T$), i.e. the number of tests needed with an exhaustive search before convergence to the optimal values. The final result is to set the crossover rate for the problem to $p_c = 0.85$.

4.3. Mutation rate $p_m$

At each generation each gene (one of the values forming a chromosome) in the chromosome will be replaced with a probability $p_m$ by a new randomly chosen value. Different solutions for $p_m$ have been tested in the range $[0.001, 0.01]$. Figures 2 and 3 (plots on the right) show the GA performance for the different values of $p_m$ in terms of number of tests needed before convergence to the optimal values (obtained with an exhaustive search). The final result converged to a mutation rate $p_m = 0.01$.

Further settings for the GA are:

- Population representation: An integer representation is used for each chromosome, where each integer corresponds either to the identifier of a node in the network or to a null value, corresponding to the situation where the number of sources is decreased by one. Each possible
solution (chromosome) is represented by a sequence of $N$ ($N$ is the number of nodes) integers, with the obvious condition that a single presence for a node in a chromosome is allowed and that at least one gene has not a null value. Each node is, therefore, identified by a unique integer $id$ from 1 to $N$.

- **Replacement scheme**: At each generation, the population, made by the parents (the old solutions) and the children (the new solutions), needs to be reduced to $\mu$. With a probability $pr = 0.25$ the parents are replaced according to a selector scheme, other values for $pr$ in the range $[0–1]$ have also been tested, but with worse results. The *RouletteWheelSelector* has been used to select the best performing chromosomes at each generation and replace the parents with the children if this was the case. The main idea behind this selector scheme is that better individuals get higher chance [32].

- **Termination factor**: The GA will terminate upon convergence. At least 20 generations are run and, in case of no convergence, the GA will terminate when the maximum generation number is reached.

- **Generation number**: The number of generations is the maximum number of tests, which leads also to the maximum time invested, in case of no convergence. The maximum generation number was set to 200.

### 5. PERFORMANCE EVALUATION

This section presents the performance analysis for the proposed solutions. In subsections ‘RLNC in addition to cooperation’ and ‘Analysis on the number of sources’, attention will be devoted to the enhancements introduced by the network coding technique used in addition to ‘classic’ cooperation in the file-sharing scenario. In this case, the set of source nodes for the cluster is elected through an exhaustive search in the solution domain. The best set of source nodes is selected after ranking each possible solution according to the correspondent service time or alternatively the average per node energy consumption. Different considerations are made when either the service time or the energy consumption is the main parameter to be minimised as the chosen set of sources will be different. As the exhaustive search is a very computational demanding solution, it is clearly not a feasible possibility for real environments, especially in case of a large number of nodes in the networks. Moreover, the results will show that no *a priori* unique choice is possible for general usage. In subsection ‘The GA performance’, the focus will be on the GA performance and the enhancements provided regarding the time spent for the cluster-head election.

In the reference scenario, an ideal medium access control (MAC) protocol is assumed on the WLAN link, without collisions and with all contending nodes having the same probability to access the medium. Although aware of the fact that this simplification might influence the final results, the proposed solutions remain valid and similar well-performing for a real MAC implementation. The file to be shared is split into $K$ chunks. If $b$ is the rate at which the nodes transmit their chunks over the WLAN link (5.5 Mbps), then the unit of time $T$ is assumed equal to the time needed to transmit the entire file at rate $b$. As for the cellular links, different possibilities are considered, namely GPRS (42 Kbps), Enhanced Data rates for GSM Evolution (EDGE) (120 Kbps), Universal Mobile Telecommunications System (UMTS) (360 Kbps) and HSDPA (1200 Kbps).

Regarding the energy consumption parameter, besides the cellular power consumption (0.4 W), the power consumption for WLAN communications in the cluster is also introduced. Three different possible states (with relevant power consumption values) are considered: transmission (1.6 W), reception (1.3 W) and idle time (0.9 W). The data rate and power consumption values used in this paper are taken from tests made on commercial mobile phone devices (Nokia N95) in order to have realistic figures in the simulated scenario.

Once the sources have been elected, the network will use the information of the cellular throughput level towards each source to proportionally evaluate the number of data chunks each source has to download; this has been noticed to improve the overall throughput performance of the cluster. The cellular download time is computed proportionally to the $b$ value and multiplied by the number of chunks each source receives.

#### 5.1. RLNC in addition to cooperation

Sample network configurations are considered to assess the actual performing behaviour of a joint use of cooperation and network coding features in a wireless cluster. Initially, the focus is on the *service time* as the objective function; it will be shown that energy consumption gains are also achievable in this case. Gains are obtained by comparing the results to the case in which each node acts ‘in isolation’ and does not implement either cooperation or network coding.

A 12-node sample network is first considered, in which all the nodes have the same kind of cellular links; *four* different cellular links are considered, GPRS, EDGE, UMTS and HSDPA. Later, the heterogeneous cases are addressed, where two different cellular systems cover either 1/4, 1/2
or 3/4 of the nodes. In the cited study cases, the optimal sub-set of sources is found through exhaustive search.

Focusing on the homogeneous scenarios in Figure 4, the obtained service time optimal values when the nodes cooperate (14.1T, 7T, 4.4T and 3.5T), compared to those obtained in the standalone case (13.1T, 4.6T, 15.3T and 4.6T), respectively, testify a gain of 89, 85, 71 and 24 per cent for GPRS, EDGE, UMTS and HSDPA scenarios. The most relevant gains from cooperation are obtained for lower values of the data rate over the cellular links (e.g. in the GPRS case). By comparing the ‘only cooperation’ scenarios to the scenarios where the network coding technique is also implemented, an additional gain of 5, 10, 15 and 20 per cent for GPRS, EDGE, UMTS and HSDPA scenarios, respectively, is achieved. The performance augmentation from ‘cooperation’ to ‘cooperation with network coding’ can be observed in all the simulated scenarios; this is an expected result, because it is related to the short-range transmissions only. This also leads to an increasing percentage gain for higher values of cellular data rates, when the time spent on the short-range transmission is proportionally larger with respect to the time spent on the cellular link.

For the heterogeneous scenarios in Figure 4, let us focus on the following sample cases in which network coding is used in addition to simple cooperation: case A: 1/4 HSDPA nodes and 3/4 GPRS nodes; case B: 1/2 HSDPA nodes and 1/2 GPRS nodes; case C: 3/4 HSDPA nodes and 1/4 GPRS nodes. A 98 per cent gain is achieved by a GPRS node in all the cases, while 42, 37 and 40 per cent gains are obtained by a HSDPA node in cases A, B and C, respectively. By comparing these values to the ‘only cooperation’ scenarios, a gain of 26, 19 and 21 per cent is the consequence of the network coding technique, for all nodes, in A, B and C cases, respectively.

As the solutions guaranteeing the best service time is chosen, it is important to observe also the consequences on the energy consumption. A distinction should be made at this point between a node elected as source for the cluster and the other nodes. The first kind of nodes additionally consume energy for the cellular link downloading. In case of cooperation, an energy gain of about 86, 75 and 43 per cent is observed for the GPRS, EDGE and UMTS nodes, while only for HSDPA nodes, there is no noticeable energy consumption benefit. For the non-source nodes instead, higher gain values are obtained; about 94, 83 and 51 per cent for the GPRS, EDGE and UMTS nodes, respectively, but still no gain for the HSDPA nodes. When network coding is used in addition to cooperation, the consequence is an additional energy consumption gain which ranges between 8 and 18 per cent in the different considered scenarios.

5.1.1. Selecting the less energy consuming solution. Different results are obtained when the solution guaranteeing the lowest energy consumption is chosen. As network coding always enhances the performance, only results when it is used in addition to cooperation are shown. Again two cases will be separately analysed: the case of a ‘source node’ and the case of a ‘non-source node’. Test scenarios are the same as in Section ‘RLNC in addition to cooperation’.

When the solution guaranteeing the lowest energy consumption is selected, the results achieved in the homogeneous scenarios do not depend on the cellular links of such nodes. This is an expected behaviour, since node positions in the network for the study cases are the same for all scenarios, and the packets’ exchange over the short-range links is similar. The average energy consumption for the non-source nodes...
source nodes always has a similar value for all the solutions, with a corresponding energy gain, compared to the non-cooperative behaviour, equal to 95, 85 and 55 per cent for GPRS, EDGE and UMTS scenarios, respectively (no gain is obtained for HSDPA nodes). For what concerns the source nodes, after selecting the best solutions, they have a lower energy consumption gain equal to 71, 68 and 36 per cent for GPRS, EDGE and UMTS scenarios, respectively. Comparing these solutions to the optimal results computed according to the service time optimisation, as shown in Section ‘RLNC in addition to cooperation’, there is an increased energy gain; notwithstanding, a worse energy gain for the non-source nodes and a worse service time gain are observed. In fact, 75, 78, 67 and 39 per cent service time gain values for GPRS, EDGE, UMTS and HSDPA scenarios are now respectively obtained.

When the solution guaranteeing the lowest energy consumption for the source nodes is chosen, the solution takes into account both the short-range and the cellular link energy consumptions. Different from the previous solution (which selected the best solution according to the energy consumption for non-source nodes), results are similar to those obtained in the case of service time optimal solution. The average energy consumption for the non-source nodes registered, respectively, a 94, 84 and 53 per cent energy gain for GPRS, EDGE and UMTS scenarios (no gain is obtained for the HSDPA scenario) when compared to the non-cooperative behaviour. By looking, instead, at the source nodes, it is observed that the energy gain now reaches 86, 75 and 43 per cent values for GPRS, EDGE and UMTS scenarios, respectively. For what concerns the service time gain, 90, 86, 75 and 38 per cent values are obtained for the GPRS, EDGE, UMTS and HSDPA scenarios, respectively.

By comparing the achieved results, the reader may observe that the ‘optimal energy consumption solution’ for the non-source nodes implies a worsened performance for what concerns the service time gain and the energy consumption gain of source nodes. For this reason, even if a very small energy consumption gain is obtained for the non-source nodes, this is not enough to justify the usage of this solution. By focusing, instead, on the ‘optimal service time solution’ and the ‘energy consumption for the source node solution’, many similarities emerge. A very reduced improvement in the energy consumption gain is attainable by using the optimal solution for the energy consumption of the source nodes, together with a very small worsening in the service time gain. For these reasons, searching for the optimal solution in terms of service time seems to be the best general option for any further analysis.

5.2. Analysis on the number of sources

This section presents an analysis on the number of nodes elected as sources in the different test cases discussed in the previous sections.

Curves in Figure 5 refer to the optimal service time solution and have been obtained with an average number of sources equal to 12, 12, 11 and 10 for the GPRS, EDGE, UMTS and HSDPA cases, respectively. The optimal number of sources tends to decrease with the increase in the cellular data rate. It must be pointed out that such a behaviour is more noticeable for larger clusters, as it will be shown in Section ‘Genetic algorithm versus random selection’. When the cellular throughput is low, such as with GPRS and EDGE technologies, it is useful to maximise the number of sources. Differently, this statement does not hold when nodes have high cellular throughput (like with UMTS

Copyright © 2010 John Wiley & Sons, Ltd.

DOI: 10.1002/ett
and HSDPA links) and when in the presence of heterogeneous scenarios.

With reference to the optimal results in terms of energy consumption for the non-source nodes in all the homogeneous cases, the best average number of sources has been computed to be around six. Noteworthy, these solutions are obtained considering the network coding technique used in addition to cooperation. For the simple cooperation solution, besides a 17% higher energy consumption for the non-source nodes, a lower value of the number of sources is obtained, which in many cases can even be equal to only one node. Consequently, a very high energy consumption is required to this single source node.

With reference to the optimal results in terms of energy consumption for the source nodes, the number of sources follows a similar trend to the optimal service time case. The best average number of sources to be elected is equal to 12, 11, 10 and 8 for GPRS, EDGE, UMTS and HSDPA scenarios, respectively.

Presented results highlight that in some cases it is not possible to have a priori knowledge of the number/type of sources required. From the study conducted it emerged, for example, that in a 12-node network with homogeneous GPRS or EDGE cellular links, the best choice is to choose all nodes as sources for the cluster. Unfortunately, this rule is not true for different cellular configurations, especially for larger networks and for heterogeneous scenarios. Therefore, the optimal solution in different considered networks shall be found through exhaustive searches with a very high number of tests (increasing with the number of nodes in the network), which is not a feasible solution. Therefore, although the joint use of cooperation and network coding has been proven to be very effective, in some highlighted conditions, it is not possible to find the best suitable solution due to complexity limitation in the best choice of the sources. This limitation is overcome by the GA as addressed in the remaining part of this section.

5.3. The GA performance

The GA algorithm can be designed to work well for both service time and for energy consumption minimisation. The fitness value of every single solution will be either the time needed to spread a given amount of data across all the network nodes or the average per node energy consumption in the delivery of all data to every node. At the end of the evolution, the best solution will give the number and the combination of nodes to be used for the file sharing service (for the remaining part of the file).

The first observed result is the capability of the GA to evolve towards a solution which is very close to the optimal one, as already demonstrated in Reference [6]. In some cases, the solution found is even equal to the optimal solution obtained with the exhaustive search. Given that this is an important result, it does not yet give information about the real benefits of using GA instead of an exhaustive search. The key parameter to be studied is the actual number of tests the GA will perform in the evolution towards the best solution and consequently the time spent during this evolution. The advantage is indeed the time needed to evolve towards it, which is severely reduced compared to an exhaustive search which tests all the possible solutions (4095 for a 12 nodes network).

In order to give the reader a clear understanding of this trend, in Figure 6 a sample 12-node case with all HSDPA cellular links is plotted; similar trends have been obtained for any other case. The number of tests and their fitness value are plotted for both the GA and the exhaustive search; the difference in the number of tested solutions is manifest. Noteworthy, a sample network with 12 nodes has been considered; increasing the number of nodes will obviously make the solution domain grow and the number of solutions to be tested increase.

5.3.1. Genetic algorithm versus random selection. The capability of the GA to converge to either an optimal or a satisfactory sub-optimal solution in a small number of tests permits to affirm that the exhaustive search of the optimal solution is not a convenient option. In this subsection, the objective is to give a new term of comparison to the GA solution performance, by considering the results achieved through a pure random choice of the sources. Solutions will randomly be tested for a time interval which is equal to the time spent by the GA tested solutions.

The random algorithm will not test the same number of solutions as the GA. This is basically due to the GA feature to evolve towards better solutions (less time demanding solutions, because closer to the best ones) and the opportunity to handle more information data. This is observed in Figure 7, where the tests performed during an evolution phase for the GA and the random algorithm are plotted for a given 12-node network with mixed cellular links. The trend of the GA is to converge versus better solutions (with smaller values of time units), and the observed throughput gain for the GA is of about 21% with respect to the random algorithm during this first phase.

Let us now consider the sample network configurations analysed in previous sections. When comparing the best GA

DOI: 10.1002/ett
solution to the best randomly chosen solution, an average throughput gain of 5 per cent is obtained. Apparently this average gain is not very large, but the reason for this is strictly related to the distribution of the fitness values of all the possible combinations in the solution domain. Only a few solutions are very bad solutions, representing peaks to be avoided. On the contrary, there are not very significant positive peaks to converge to, and the distance of many solutions from the optimum is not very large. Likely, by increasing the number of nodes, this gain will increase as the distribution of the fitness values will be influenced by the larger network size.

5.3.2. Genetic solution versus network size. Results for larger networks (up to 28 nodes) are presented in this last part of the research work. Logically, the exhaustive solutions for larger networks are not shown, as these are very high computational demanding. But, since the GA proved to be able to guarantee a very good solution, the analysis is based on the GA results, which are obtained in very short time intervals.

Sample cases are considered for GPRS and HSDPA homogeneous scenarios, and for a heterogeneous scenario, where half of the nodes are equipped with either GPRS or HSDPA cellular interfaces. Figure 8 highlights that the GPRS scenario uses all nodes as sources in all the cases, while for the HSDPA scenario, the percentage of nodes acting as sources is decreasing for larger networks. This interesting behaviour is even more evident in the heterogeneous scenario, where the best solution is to exploit the fast
HSDPA nodes for downloading data over the cellular link instead of involving all the slow GPRS nodes as well.

For what concerns the service time, refer to Figure 9. It is possible to notice how for the GPRS scenario this value is decreasing with the number of nodes. This is an expected result, since all the nodes act as sources and, consequently, the time to download the data is decreasing proportionally to their number. In the other two considered cases, the time spent on the cellular link is actually smaller than the short-range transmission time. For this reason, the overall service time increases with the number of nodes in the network.

For the energy consumption parameter, a difference should be made between source nodes and non-source nodes. For the source nodes the energy consumption is following a plot which is similar to the service time plot, while for the non-source nodes an energy consumption which is increasing with the network size is observed.

6. CONCLUSIONS

This research work analysed a cooperative file-sharing service in a WLAN-based cooperative cluster where RLNC is implemented to enhance performances. The improvements introduced by the network coding technique are analysed, by showing how both service time and energy consumption values are benefiting from its usage in addition to the wireless cooperative communication.

In such a scenario, where nodes may differ in available cellular throughput and in their position into the cluster, a good choice of nodes acting as sources for the cluster is of utmost importance. To this aim the use of a GA is proposed and, being the domain of the possible solutions very large, this technique proved to fit very good to this purpose. The GA performance in different cellular link configurations has been assessed and its good behaviour has been proven when focusing either on the service time and, alternatively, on the energy consumption optimisation, and when comparing it with the exhaustive and random search algorithms for selection of sources.

REFERENCES


Copyright © 2010 John Wiley & Sons, Ltd.