A Cooperative File Downloading Scheme with Genetic Algorithm

Xing Zhang, Lanlan Rui
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, China
E-mail: {holy198910,llrui}@bupt.edu.cn

***Abstract*—Existing cooperative file downloading schemes have some blindness on the selection of collaborative nodes. In this paper we present a cooperative file downloading scheme with genetic algorithm to resolve the collaborative nodes selection problem, especially in the case that we cannot have a prior knowledge about the data rate and position of the mobile nodes. A simple and efficient extended on-demand proxy discovery algorithm will be used to find the potential cooperative nodes. Character of the genetic algorithm (GA) makes it suitable for the selection of cooperative nodes. After this, client uses multiple parallel paths for file downloading. The simulation results show that the cooperative file downloading scheme with genetic algorithm can successfully select collaborative nodes with better performance and effectively reduce file download latency.**

Keywords—Genetic Algorithm;Wireless Cooperation; MANETs; Cooperative File Downloading

I. INTRODUCTION

Recently, more and more research work is focused on using wireless cooperation to improve the latency of cellular data networks. And it has already been proved that using wireless cooperation to get the same resources relative to accessing external server can save more time and energy[1, 3]. So how to find the collaborative nodes and realize cooperation among them will be the key point to improve the latency.

In [5], inspired by the *Cooperative ad Hoc network to sUpport Messaging* (CHUM) project [4], the authors formed the idea of utilizing idle cellular link of other nodes in MANETs and proposed a cooperative parallel file downloading scheme. The use of on-demand proxy discovery algorithm and parallel request can surely reduce the latency. But the selection of proxy nodes only based on hop count and round trip time which is estimated in proxy discovery process, may not make sure to get the proxy nodes with better performance, especially when nodes have different data rate. Here the better performance means smaller latency from downloading to transmitting the file portion to the client.

On the basis of the work in [5], in this paper, we propose *the cooperative file downloading scheme with genetic algorithm*, which is aiming at tackling the collaborative nodes selection problem, especially in the case that we cannot have a prior knowledge about the data rate and position of the mobile nodes. We assume that mobile devices have the ability to access the cellular data network and IEEE802.11b-based MANET simultaneously. The base station is not involved in cooperation. The file on the content server is split into a plurality of segments. When a terminal (as client) wants to download the file, it will select several neighbor nodes with better performance acting as proxy nodes to complete file downloading together. The proxy nodes will download different parts of the file via their idle cellular link and transmit them to client via the IEEE802.11-based wireless link. As is shown in Fig.1, some mobile users are in same MANET with their cellular link connecting to the internet. Client A wants to download the file on the content server, after broadcasting the proxy discovery request, A gets that B、C、D and E both have the willingness to participate in cooperation. After selection, B、C and D are selected as proxies to download the file.

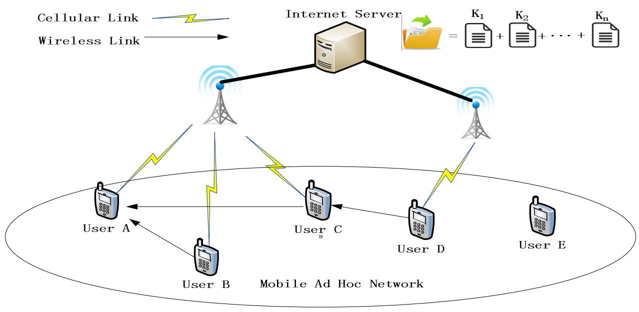

Fig. 1. Cooperative File Downloading with GA

II. RELATED WORK

As an intelligent optimization algorithm, genetic algorithm has been widely used in many areas. In [9], GA is applied to selecting the energy efficient clusters header in wireless sensor networks. On the selection of cluster header, performance comparison between GA and Exhaustive Search, and the comparison between GA and Random Selection Algorithm can be seen in [2,8]. In [10], aimed at defining the number of clusters in network and allocating users to different clusters, the authors use GA to design the topology of the LAN.

There are also many works on the basis of the integration of ad hoc and cellular data network. In order to improve data availability and access efficiency of local resources, a novel cooperative caching scheme for on-demand data access was proposed in [6]. In [3], the author proposed the conception of *Collaborative Urban Computing* and *Self-Organized Market*

Algorithm to support serendipitous cooperation between a set of users physically located in an urban environment, but limited data rate and transfer range of Bluetooth are the bottleneck of the system.

III. COOPERATIVE FILE DOWNLOADING SCHEME

The cooperative file downloading scheme mainly contains three procedures: the discovery of potential proxy nodes, the selection of proxy nodes and file downloading. AODV [7] routing protocol is used to establish the routing table from client to proxy. In discovery phase, the on-demand proxy discovery algorithm mentioned in [5] is used to find potential proxy nodes set, but we will make some extension to make it suitable for our need. After downloading a small test part of the file, GA will be executed to realize the selection of proxy nodes. At last the file downloading with parallel paths will be started.

A. The Potential Proxy Discovery

In this phase, unlike in [5], the *proxy discovery request* (PDREQ) message will not carry any information of the file size. This is because the size of the file that each node needs to download will be determined by utility. The utility will be introduced in next section and now we will make a brief introduction about the on-demand proxy discovery algorithm in the following part.

The client broadcasts the PDREQ message to its neighbors, the PDREQ carries the client's address (SR), request identification (REQID) and request broadcast range (RBR). The SR and REQID can be used to uniquely identify the request. The REQID is used to distinguish the freshness of this request, which is increased every new round of the proxy discovery. The function of RBR is to limit transmission range of the request and it will be decreased every time the message is rebroadcasted. The *proxy discovery reply* (PDREP) message is used by the node with willingness to participate in cooperation. The PDREP carries the REQID, the address of the proxy (PR) and the reply identification (REPID). The REPID is used to distinguish the freshness of the reply, which is increased every new round of proxy discovery.

The routing table from client to proxy is established by AODV protocol. The PDREQ and PDREP message are used in same way as the route request and route reply in AODV. When mid nodes in MANETs receive the PDREQ, they will update the routing entry for client. And if the REQID value of the received PDREQ is smaller than or equal to which it holds, the node will drop the PDREQ. Otherwise it means that the request has not been processed. The node will update the REQID value it holds and decide whether or not to participate in cooperation based on its battery level or current network traffic load and other factor. During the discovery process, if RBR is positive, the mid node will decrease the value of RBR and rebroadcast the request to all its neighbors. If the node wants to participate in cooperation, it will send a PDREP message to the client. When mid node receives the PDREP, the comparison of the REPID value is also unavoidable, a new routing entry is inserted into routing table for the proxy. If client receives the PDREP, it will record the address in a prepared set $\mathcal{J}$. Fig.2 shows an example of the proxy discovery process.

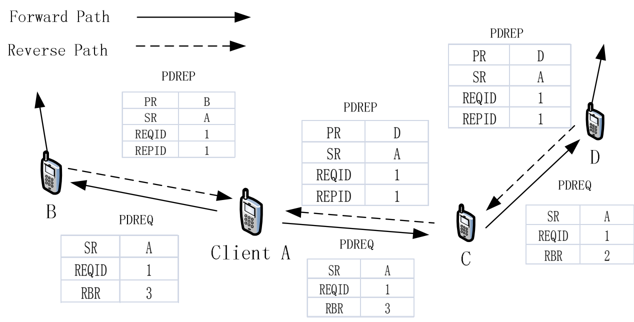

Fig. 2. Examples of Proxy Discovery Process

B. Proxy Selection using GA

It is obvious that the time needed to transmit a given amount of data to the client is influenced by the cellular link capability and the position of proxy in MANET. So the main purpose of this phase is to get the combination of proxies either with higher data rate or with less hop count to client. The reason to use GA is that, when the size of $\mathcal{J}$ is larger, rapid astringency and the expandability of genetic algorithm make it suitable for our need.

After getting set $\mathcal{J}$, the client will send the *file portion downloading request* (FDREQ) to all nodes in $\mathcal{J}$. The FDREQ includes the location of file, the bytes-range of file portion, the client's address, and the request identification and a flag. In this phase, flag equals to zero, it means that, the FDREQ is used for collecting information of proxy. After downloading specified length of the file, proxy should return a *file downloading reply* (FDREP) to the client. FDREP contains t_{ja} which represents the time needed for downloading the specified file portion, the proxy's address and battery level. Besides the aforementioned information, when receiving the FDREP, client will also record the delay t_d from proxy to client.

A timeout value will be associated with FDREQ, indicating the time period during which FDREP must be received from nodes. After receiving replies, client will execute the genetic algorithm. Details about the genetic algorithm are summarized in the following:

1) Initialize Population: The number of population is $\mathcal{N}$, which equals to the number of nodes in set $\mathcal{J}$. Every chromosome represents one possible solution with a sequence of n integers, n is unique and $n \in [1, \mathcal{N}]$, each integer corresponds to the identifier of a node in set $\mathcal{J}$ and will only appear once in a chromosome.

2) Evaluation: The fitness value of each chromosome is the maximum time needed to transmit the file portion to client. The evaluation procedure can be divided into following three steps.

a) Calculate the utility value u_{ji} of each proxy in $\mathcal{J}$:

$$u_{ji} = w_s \cdot s_{ji} + w_{bl} \cdot bl_{ji} \quad (1)$$

Where s_{ji} indicates the data rate and is calculated by t_{ji} and the length of the test file ℓ, $w_s \in [0,1]$ indicates the weight of data rate, bl_{ji} represents current remaining battery level,

$w_{b\ell} \in [0,1]$ indicates the weight of terminal's remaining battery level, $w_{b\ell} + w_s = 1, w_{b\ell}, w_s > 0$.

b) Calculate the size of file that each node needs to download based on the utility value u_{ji}:

$$\mathcal{L}_{ji} = round\left(\mathcal{L} \cdot u_{ji} \Big/ \sum_{i=1}^{i=n} u_{ji}\right) \qquad (2)$$

Where $\mathcal{L}$ represents length of the file to be downloaded, n is the number of nodes in this chromosome.

c) According to s_{ji}, $\mathcal{L}_{ji}$ and t_d, estimate $\mathcal{T}_{ji}$ of each terminal, $\mathcal{T}_{ji}$ represents the time needed from downloading to transferring the file portion to the client node. Then select the max $\mathcal{T}_{ji}$ as the fitness value of current chromosome.

3) Crossover: With the crossover rate pc, when the value is less than pc, two chromosomes will use the one-point crossover solution to produce two new chromosomes.

4) Mutation: Each gene in the chromosome may be replaced by other randomly selected reasonable values with the probability pm.

5) Elite Strategy: In order to improve the algorithm's efficiency, the elite strategy was introduced to keep the optimal individual to the next generation.

6) Termination factor: the number of generations Gn is the terminating factor for the GA.

At this point, the selection of the proxies is over. According to (1) and (2), the size of the fragment each node needs to download is proportional to the utility value it has. Nodes with higher utility will download more. So if the eventual selected proxies contain the node with low data rate, the file portion it downloads will also be small. This makes sure that the node with low data rate will have small influence on the delay.

C. File Downloading

When completes the selection, the client will send the FDREQ to the proxy nodes with a non-zero flag value. The portion size is determined by (2) during the selection process. After getting the request, the proxy terminals will use the cellular data link to download the file portion, and then transmit it to the client via the multiple wireless paths.

In order to avoid the idle time during which no portion is downloaded, when the client is downloading, the proxy discovery and selection will be proceeding simultaneously. After receiving the fragments from the proxies, a new round of proxy discovery and selection will start.

Due to the mobility of the nodes, the established routes may be broken. The failure recovery used in [5] can also be used here to detect and recover the route failure.

IV. PERFORMANCE EVALUATION

A. Simulation Setup

The simulation is based on the ns-3 simulator which is widely used in networking research recently. 75 mobiles nodes are random placed in the area of 500 m * 500 m, the node mobility pattern is the random walk 2d model. Each node is equipped with a cellular interface and an IEEE802.11b interface which supports 11Mbps data rate at the 100 meter range. The AODV that included in the ns-3 simulator is used to establish the routing table from the proxy to client.

B. Simulation Results

In the simulation, proxy request hops is limited to 3. The number of the potential proxies varies from 3, 6, 9, 12 to 15. The cellular bandwidth varies from 128Kbps, 256Kbps, 512Kbps to 1Mbps. In order to simulate the real scenario, mixed data rate is also considered. In following pictures, network Mixed A represents that half of the nodes with the data rate at 1Mbps and the other with the data rate at 256Kbps, network Mix B represents that one third of the nodes with the data rate at 256Kbps, another third with the data rate at 512Kbps, the rest with the data rate at 1Mbps.In the following pictures, the performance gain is relative to the downloading case where no cooperation exists.

Due to the role of the genetic algorithm, the parameters of the genetic algorithm will be determined firstly. As is shown in Fig.3, with the specified generation number, when $pc = 0.8$ and $pm = 0.1$, the genetic algorithm always fast converge to the better solutions namely the average performance gain relative to downloading without cooperation is always larger than others. Besides the combinations listed, other combinations have also been tested, and with worse result too.

The impact of the size of the file portion is shown in Fig.4, when the number of the proxies equals to 5, the optimum value of $\mathcal{L}$ for this simulation scenario is between 150 to 200. If the file portion is small, according to (2), the size of the file each proxy needs to download will also proportionally decrease. The time spent on routing will be more. Larger portion does not provide enough modularity for parallel downloading, and also results in performance degradation.

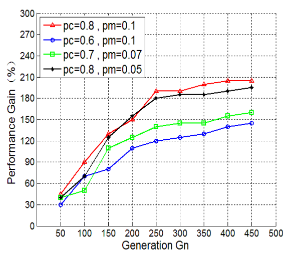

Fig.3. Parameter of Genetic Algorithm

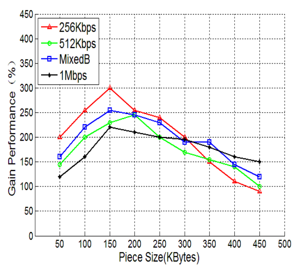

Fig.4. Parameter of Genetic Algorithm

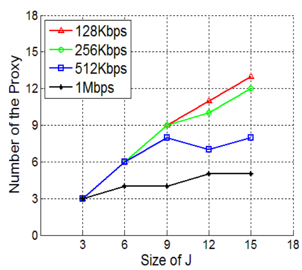

Fig .5. (a) Same Cellular Link Bandwidth

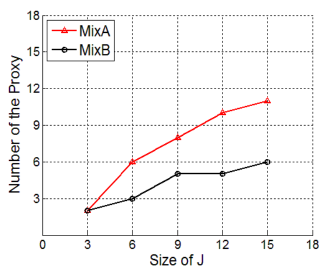

Fig.5. (b) Mixed Cellular Link Bandwidth

And next, we will observe the number of the proxies under different size of $\mathcal{J}$ and data rate. When the size of $\mathcal{J}$ varies from 3, 6, 9, 12 to 15, with the moving speed 5m/s, the number of the selected proxies under different cellular link bandwidth is shown in Fig.5(a) and Fig.5 (b).

When the nodes in MANETs with the same low cellular link bandwidth like 128Kbps, 256Kbps or the combination of the both like network A in Fig.5 (b), the number of the proxies is always close to the size of $\mathcal{J}$. Thus because with all nodes downloading the fragment, according to (2), the size each node downloads will reduce proportionally, so the time spent on the cellular link is reduced correspondingly. With higher cellular link bandwidth like 512Kbps, 1Mbps, this phenomenon is not evident anymore. Relative to the time spent on the cellular link, the time spent on the transmission of the fragment via the wireless link will be larger. In the both aforementioned situations the GA is more likely to choose the proxies with less hop count to the client. But in the network B which is shown in Fig.5 (b), with the mixed data rate, one half nodes with data rate at 1Mbps and one half nodes with data rate at 256Kbps, the results show that GA always chooses the node with higher data rate and avoids involving the nodes with lower date rate.

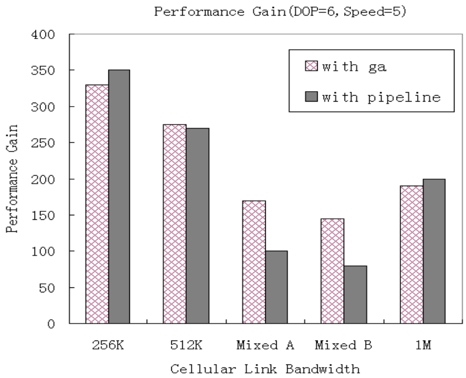

Fig.6. Performance Gain Comparison

In Fig.6, the X axis represents the cellular link bandwidth. The Y axis represents the performance gain relative to the situation where no cooperation exists. With the degree of *parallelism (DOP) always equals to 6 and the number of the potential proxies equals to 12. From the figure, we can see that, in the case that nodes have the same bandwidth, the two mechanisms have similar performance. But in general case that the proxies with different bandwidths, the scheme with GA has a better performance than the parallel cooperative file downloading scheme.*

V. CONCLUSIONS

In order to tackle the cooperative nodes selection problem under the condition that we cannot have a prior knowledge about the data rate and the position of the nodes, in this paper, we propose the Cooperative File Downloading scheme with genetic algorithm. The simulation result shows that the Cooperative File Downloading scheme with genetic algorithm is always performing better on reducing the file downloading latency than the cooperative parallel file downloading scheme.

ACKNOWLEDGMENT

This work was partially supported by National Natural Science Foundation of China (61302078, 61372108), Funds for Creative Research Groups of China (61121061), Beijing Higher Education Young Elite Teacher Project (YETP0476).

REFERENCES

[1] Lehtomäki, J. J., et al. "Direct communication between terminals in infrastructure based networks."Proc. ICT-MobileSummit (2008): 1-8.

[2] Militano, Leonardo, et al. "A genetic algorithm for source election in cooperative clusters implementing network coding." Communications Workshops (ICC), 2010 IEEE International Conference on. IEEE, 2010.

[3] Bojic, Iva, et al. "Collaborative urban computing: Serendipitous cooperation between users in an urban environment." Cybernetics and Systems 42.5 (2011): 287-307.

[4] Zhu, Danyu, and Matt Mutka. "Sharing presence information and message notification in an ad hoc network." Pervasive Computing and Communications,

[5] Zhu, Danyu, Matt W. Mutka, and Zhiwei Cen. "Using cooperative multiple paths to reduce file download latency in cellular data networks."Global Telecommunications Conference, 2005. GLOBECOM'05. IEEE. Vol. 5. IEEE, 2005.

[6] Du, Yu, Sandeep KS Gupta, and Georgios Varsamopoulos. "Improving on-demand data access efficiency in MANETs with cooperative caching." Ad Hoc Networks 7.3 (2009): 579-598.

[7] Ad-hoc On-Demand Distance Vector Routing, Perkins, Charles E., and Elizabeth M. Royer. "Ad-hoc on-demand distance vector routing." Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on. IEEE, 1999.

[8] Militano, Leonardo, et al. "Network coding and evolutionary theory for performance enhancement in wireless cooperative clusters." European Transactions on Telecommunications 21.8 (2010): 725-737.

[9] Hussain, Sajid, Abdul Wasey Matin, and Obidul Islam. "Genetic Algorithm for Energy Efficient Clusters in Wireless Sensor Networks." ITNG. 2007.

[10] R. Elbaum and M. Sidi, "Topological Design of Local-AreaNetworksUsing Genetic Algorithms." IEEE/ACM Transactions onNetworking,oct1996,vol.4