2011 IEEE/ACM International Conference on Green Computing and Communications

Time-slot Energy-efficient Scheduling Algorithm for Capacity Limited Networks

Chenying Hou
Institute of Computing Technology
Graduate University
Chinese Academy of Science
houchenying@ict.ac.cn

Fa Zhang
Institute of Computing Technology
Chinese Academy of Science
zf@ncic.ac.cn

Lin Wang
Institute of Computing Technology
Graduate University
Chinese Academy of Science
wanglin@ict.ac.cn

Zhiyong Liu
Institute of Computing Technology
Chinese Academy of Science
zyliu@ict.ac.cn

Abstract—**The current networks are often designed with redundancy in order to deal with unexpected failures, but this makes a large amount of energy consumption and bandwidth waste. In the real networks, the performance is limited by the link's capacity which gives an upper bound of the traffic amount conveying through the links. In order to explore energy saving methods in the networks, we consider a model with the link capacity constraints and study the globally energy-saving routing strategy under the capacity assumption.**

In many related studies, the traffic demands are considered to be scheduled simultaneously in one round, which provides us the lower bound of the demands routing, but the lower bound of one-time scheduling method is too relaxed and results in unnecessary link idleness and extra energy consumption.

This paper focuses on a time-slot globally energy-saving routing strategy based on the capacity limited network model. In order to increase the bandwidth utility ratio a new scheduling model with capacity constraint is proposed and a scheduling strategy is developed that can decrease the energy consumption as well as meet the performance requirement. The scheduling strategy is a time-slot energy-efficient algorithm. It splits the scheduling demands time window into more than one time slot and allocates all the demands into this two time slots with the goal of minimizing energy consumption. Experiment results show that the time-slot globally routing algorithm effectively reduces energy consumption compared with existing methods.

I. INTRODUCTION

In the last decades, the Internet has evolved from a small computer network, used primarily by academics, into a worldwide communication medium with significant impact on the global economy. The widespread use of computers and the Internet is also having an effect on the global energy consumption. The energy consumption of the routing fabric of the Internet sharply increases during recent years.

In current networks, energy consumption is tremendous because of the inefficient design of network elements and routing strategies. They are often designed for the accommodating future growth, planned maintenance or unexpected failures, or quality-of-service guarantees, which are not aimed at energy optimization goal. This provides us an opportunity to save energy by energy-efficient routing.

The network energy saving methods can be classified into two types [1]: one is the network element level saving strategy, which focuses on the power management of the routers and hosts. The other one is the global level saving including the scheduling and deployment of the network traffic.

Methods to reduce the energy consumption of network elements have drawn a great deal of research interest, during past few years. Speed scaling and powering down are two promising mechanisms for dynamically adaptive energy consumption in the data-path devices of network elements according to the actual traffic load.

When considering the global level energy saving strategy, routing algorithms and network protocols are designed with the objective of energy consumption minimization. There are some routing strategies such as shortest path method, heuristic strategy and optimaization problem known as network flow problem. Several papers work on the energy saving at a global network level. Nedevschi et al. [5] studied the global energy saving under with the sleeping and rate adaptation method. In the study of the energy saving method, only a few papers work on the global wired network energy saving [2]. While the globally optimization is a more complex but energy efficient way.

Spyridon Antonakopoulos et al. [1] studied the power-aware routing with rate-adaptive network elements. They assumed that network elements are rate-adaptive whose energy cost is dependent to the traffic through the elements. Every network element has an energy function to reflect the energy-adaptive ability of element. They

978-0-7695-4466-3/11 $26.00 © 2011 IEEE
DOI 10.1109/GreenCom.2011.11

propose a heuristic greedy energy-efficient algorithm based on the energy function. Antonakopoulos et al. found that the energy saving performance differs from various graph topology and the different kinds of energy functions. Their routing strategy is simple, but does not consider the global optimization and traffic delay caused by the uncertain rounds of greedy iterations.

Mattew Andrews et al. [2] worked on a model with network edge having the speed scaling capability, characterized by an energy curve $f_e(s)$. They also proposed a routing problem with the target of routing a long-time scale traffic demands matrix with the goal of saving energy. They formulated this problem into an integer programming. In their model, the network edges in their assumption are not bounded by the capacity constraint.

In real network condition, capacity is an important parameter for network element. The current network performance is largely limited by network element's capacity constraint. The amount of traffic loaded on each edge is limited, and simply minimizing the energy consumption in network may violate the constraint of capacity and not be applicable in true network situation.

In our research, we try to find the global optimal scheduling result, and we first formulate it as an optimization problem with one-time scheduling method then we propose a time-slot scheduling method. Different from the works in some related studies, the traffic demands here is not a long-time scale traffic matrix as continuous flow, so the only one time scheduling may result in link idleness and energy waste.

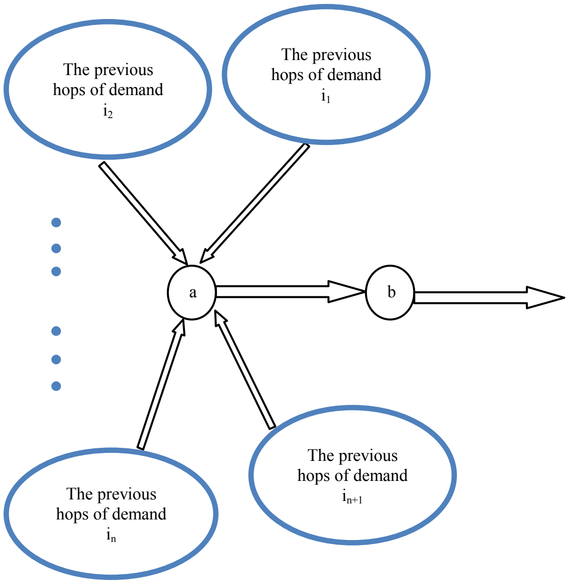

Figure 1. Routing case of edge (a, b)

Consider the situation like this, from the one-time scheduling we get the optimal scheduling solution and for an edge (a, b) we have deployed demands i_1, i_2, i_3,… i_n on it. And now the edge (a, b) is fully used and cannot be loaded by any more demands based on the energy optimal objective and capacity limit. But for demands i_1, i_2, i_3, … i_n , they may have various delays caused by experiencing different number of hops in the network and arriving at node a at different time. So, because of their various delay, the edge (a, b) is sometimes idle having bandwidth waste and not fully occupied.

Thus, in order to increase the utility ratio of the link bandwidth, we propose a time-slot scheduling method which realizes the more than one time scheduling. This paper focuses on a model in which energy using of the network elements has the rate-adaptive ability and network elements can adjust its working speed according to the traffic through it and every link has the capacity restriction. A traffic demand matrix should be scheduled within two time slots. We assume each network link has an energy function $f_e(s)$. It reflects the energy cost of the link e when conveying x amount of traffic. And we can formulate this time-slot scheduling problem into an optimization problem. We propose a new routing scheduling problem with the objective of minimizing the energy cost globally. The above problem can be described as a NLP (non-linear programming), which can be solved using existing techniques, like the existing Lingo software. Notes that with our proposed model, if the fractional routing is acceptable, then each demand can be routed via multiple paths, but meanwhile some technique must be applied to ensure the packet reordering of the fractional part [2]. If fractional routing is not allowed, restriction should be added for the integer flow. And the problem will become a mixed integer NLP which is a NP-hard problem. Because of its hardness, approximation algorithms to approximate the optimal solution may work efficiently. And the approximation algorithm has been studied in some works [2, 3] with the similar models without the capacity restriction. And the randomized rounding technique is helpful [3] when solving the integer programming in the routing problems. But this paper primarily focuses on exploring the energy-saving method and opportunity under the newly formed model while we will not discuss the approximation technique in detail so we only consider the fractional routing condition.

This paper is organized as follows. In Section II, a model with capacity constraint for energy-efficient routing is described and a strategy of time-slot scheduling is also discussed. The simulating experiment and results are introduced in Section III. And Section VI is the conclusion.

II. A MODEL FOR ENERGY-EFFICIENT ROUTING

We will describe the network topology and its components with respect to energy consumption. Our research on the network routing is based on the test instance derived from the Rocketfuel study [4]. We build a city graph according to the topology in Rocketfuel dataset. We will model a national wide network as a city graph G = (V, E), and nodes in graph represents the cities, and the edges represents the links between each city and between two cities there is at most one link(multi-edge link is not allowed).

We also assume each link has an energy cost function $f_e(s)$ which gives the energy cost of the link e when conveying s amount of traffic. And each edge in the graph has a capacity upper bound. Our goal is to route all of the demands in the given traffic matrix with minimum cost without violating the capacity constraints. It can be found that the hardness of the problem is largely depended on function $f_e(s)$, and for most function $f_e(s)$, the problem is NP-hard.

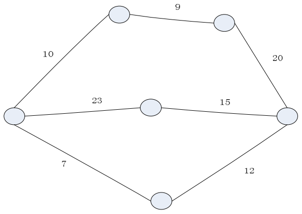

Figure 2. Network graph

In this paper, we primarily work on two kinds of energy function $f_e(s)$: the linear functions and the convex energy functions from the study of Spyridon Antonakopoulos et al. [1].

The traffic demand can be described by a traffic matrix. An entry of the matrix is t[a][b], and it represents the traffic amount from source node a to destination node b.

Link capacity is another important parameter for a network element. It gives each edge upper bound for the traffic amount on it. As for the link capacity estimation, we derive it by routing the traffic matrix in a min-hop manner on a modified graph as described [1], and the modified graph is created each time by removing one node to simulate the network failure. Then we can get the maximum load of each link tolerating one node failure. We use this method to estimate capacity since the min-hop routing is widely applied in ISPs [1]. The input of the routing algorithm will be fully discussed in the experiment part.

A. *One-time scheduling*

Based on the network topology model described in the previous part, we first study the problem of scheduling a given traffic matrix for only once which we called the one-time scheduling. The energy saving routing strategy is various from the global optimization method to the locally greedy method. Here we prefer to concern about the globally optimal solution. If the fractional routing is permitted, we can formulate this routing problem to a linear programming problem as follows (P_1). The variable $y_{i,e}$ indicates the amount of traffic which demand i allocates on link e, and x_e shows the total traffic amount on link e. Meanwhile the value of y should satisfy flow conservation of each demand in the traffic matrix.

(P_1)

$$\min \sum_{e} f_e(x_e)$$

Subject to

$$x_e = \sum_{i} y_{i,e} d_i \ \ \forall e$$
$$x_e \leq c_e \ \forall e$$
$$y_{i,e} \in [0,1] \ \forall i, e$$
$$y_{i,e} : \text{flow conservation}$$

The above linear programming can be solved in polynomial time so we can get the fractional routing solution easily, and this is also the energy optimal solution. Notice that the above model (Model P1) is different from existing models in that it considers the capacity of each network edge for when routing the demands, as expressed by $x_e \leq c_e \ \forall e$.

B. *Time-slot scheduling*

Note that in the above models, we are under the assumption that all the traffic demands are one-time scheduled. But the only once scheduling over a long period of time is not practical under some situations. It is widely known that network is an asynchronous system. In the real network, the simultaneity as described in the optimization problem above is hardly satisfied, because the transferring rate differs from one link to the other and we cannot precisely control the demand conveying process. So the one-time scheduling of the traffic demands can only bound the worst situation and guarantees no traffic congestion while it also results in link idle because delay of each demand is quite different. If we design the routing strategy under this one-time scheduling model, the link may not be fully occupied at some time which will also cost the bandwidth waste and extra energy consumption as described in the Section I.

Based on the above analysis, we propose a new scheduling model where the demands are scheduled in more than one time slots instead of one-time scheduling. For each of the time slot we will arrange some demands routed. We use a two time slots model to schedule demands. In the first time slot, some demands in the traffic matrix are scheduled, and in the second time slot, the remaining demands in the matrix are deployed in the network. After these two rounds, all the traffic demands can be served. Notice that although the demands are scheduled in two-time slots, the purpose of the two-time slot schedule is ONLY to decide which path the demand will follow. In the routing stage, the boundary of the two time slots should disappear and for any demand i, if it's given path(obtained from the scheduling stage above) is idle although the solution of demand i indicate that it should be routed in second round it can be routed through the path immediately without waiting .

We are not sure whether slots number equal to 2 is the optimal choice, and determining the number of slots implies a tradeoff between the energy-saving opportunity and the queuing delay which reflects the service performance. And the reason for fixing number of slots by 2 is that it will not cause the problem too hard while it can also be used to explore how this model will work in reducing energy and guarantee the delay not too long.

All the demands can be allocated into these two slots. And how we should allocate these demands into two time slots becomes a new problem we should deal with. The allocation strategies can be many types such as randomly choosing some demands among the traffic matrix for the first time-slot scheduling and then deal with the left demands in the second time-slot scheduling, and another method can be that the demands are divided into two parts with the equal number of demands. However, the above allocation strategies have no guarantee on whether the energy consumption of that method is optimal. In order to choose an allocation strategy with the best energy-efficient performance we propose a new model of optimization problem, so we can get the arrangement of the demands by formulating this problem as a non-linear programming with the objective of energy minimization as follows (P_2). Notice that this is a new model compared to the current existing models and methods. The problem is non-linear programming because the flow conservation expression in the time-slot scheduling assumption is non-linear function.

(P_2)

$$\min \sum_{e} f_e(x_{e,k}) \quad k = 0, 1$$

Subject to

$$x_{e,k} = \sum_{i} y_{i,e,k} d_i \ \forall e \ \ k = 0, 1$$

$$x_{e,k} \leq c_e \ \forall e \ \ k = 0, 1$$

$$y_{i,e,k} \in [0,1] \ \forall i, k = 0, 1$$

$$y_{i,e,k} : \text{flow conservation}$$

We assume that the fractional routing is permitted in each time slot, thus some fraction of a demand may be scheduled in the first time slot and the left part should be scheduled in the second, and the variable k indicates which time slot is considered and the k value can be 0 or 1, such as $x_{e,1}$ represent the total traffic amount on link e in the first time slot.

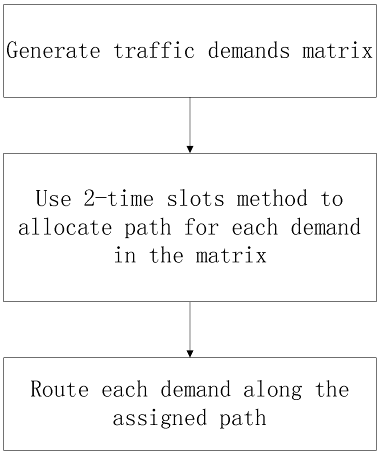

Figure 3. Routing process

From the above global optimal programming, we can get the result of $x_{e,k}$ and $y_{i,e,k}$, and definitely know which path each demand should followed. Figure 3 shows the process of the algorithm.

III. EXPERIMENT AND RESULT

We conduct the simulated experiment with the dataset from Rocketfuel study [4, 1]. In the experiment, we test the model described above with capacity constraint, and we make the comparison in energy consumption of the one-time scheduling strategy and time-slot based scheduling method. We also do the experiment of the shortest path routing algorithm without consideration on the energy. And we use the shortest path routing result to evaluate the energy saving performance of the algorithm in this paper.

Rocketfuel is an ISP toplogy mapping engine [4]. It uses the traceroutes information to construct the database. It contains many ISPs including Europe, Australia and the United States.

In our experiment, we derive a city graph from the 3967 dataset which provide us trace route information and the router connection state of the US. We preprocess the data (including building city graph, computing link capacity and deriving the traffic matrix) in the Spyridon Antonakopoulos way [1]. First, we get the topology of all the routers located in different cities from the US dataset. Then, the minor cleanup are made that if the topology contains more than one connected components, then we discard all nodes and links that are not in the largest component.

In the router graph obtained from above, a city may have many routers. So we create city graph by keeping one node representing a distinct city and a link between cities a and b exists if and only if there exists a link in the router graph with one of its endpoints located at city a and the other at b. Figure 4 presents the router graph and city graph of the Rocketfuel dataset 3967. And we do not allow the parallel links in the city graph even though many links may exist in the real router-level network.

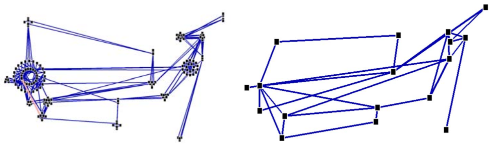

Figure 4. Router graph and city graph of the Rocketfuel AS 3967

The calculation of the traffic matrix T is based on a population-distance model, in which the traffic between two nodes is proportional to the product of their respective populations, divided by a distance factor [1] . And we suppose that each node in the router graph has a unit of population, so a node in the city graph has the population of the total routers located in that city. The reason for this assumption is that ISP deploys the right number of routers to carry the traffic generated by each city. We use the following formula to compute traffic between two cities a, b [1]:

$$\mathrm{T[a,b]} = \frac{|\mathrm{routers(a)}| \times |\mathrm{routers(b)}|}{\mathrm{e}^{\mathrm{distance(a,b)/1500}}}$$

We also need to determine capacity values for each link. In order to assign sufficient capacity for each link and also reflects the real network condition, we choose the shortest path method to generate capacity value. Each pass we make a one-node failure, and route the traffic matrix on this damaged graph until all the one-node failures have been tried, then we set the capacity of each link equal to the maximum load on that link. Different routing algorithm certainly will lead to different link capacity. The min-hop routing strategy is chosen here due to its wide application in ISPs.

The energy function for rate-adaptive network links used in the experiment is described in the above parts, and we use the linear energy function as follows and figure 5 shows the linear function curve.

$$\mathrm{f_e(x)} = \frac{\mathrm{c_e} - \sigma}{c_e} \cdot x \quad 0 \le x \le c_e$$

And the convex energy functions as follows:

$$\mathrm{f_e(x)} = \frac{\mathrm{c_e} - \sigma}{c_e^2} \cdot x^2 \quad 0 \le x \le c_e$$

In the above energy functions, $\sigma = \beta \cdot c_e$ among which we only consider the situation when $\beta = 75\%, 50\%$. And figure 6 shows the convex energy curve.

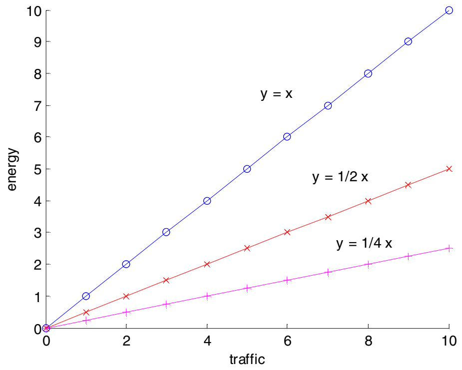

Figure 5. Linear energy function curve

In our experiment we assume that fractional routing is legal and we obtain the optimal solutions from lingo solver. For each experiment, we perform the routing algorithm with different energy function, and the value of β varies from 0 to 75%. The experiment result is shown in table 1. In table 1, each data represent the total energy consumption of the city graph under different energy functions.

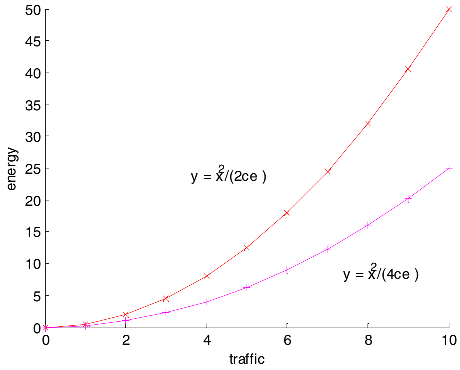

Figure 6. Convex energy function curve

The traffic matrix of 3967 dataset is large of about 300 units of demands. And the Lingo solver only gets the local optimal solution for the 2-time slot method. But the local optimal result also shows that it reduces energy consumption effectively.

From the experiment result we can observe the following points:

First, the time-slot scheduling method saves more energy when routing the same traffic demands matrix compared with the one-time scheduling method algorithm and the shortest path routing algorithm.

	Shortest path	one-time scheduling	Time-slot scheduling
$f_e(x) = x$	45848	45848	45848
$f_e(x) = \frac{1}{2} c_e x$	56198419	47905680	46000000
$f_e(x) = \frac{1}{4} c_e x$	28090610	23945050	22900000
$f_e(x) = \frac{x^2}{2c_e}$	20563	12947	9410
$f_e(x) = \frac{x^2}{4c_e}$	10205	6474	4710

Table 1. 3967 total energy consumption

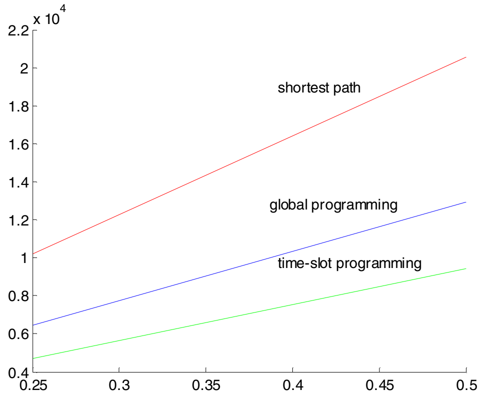

Figure 7. Convex energy function total energy consumption

Second, the one-time scheduling method and the time-slot scheduling method are both energy effective and perform better in energy saving compared with shortest path algorithm under most conditions. For some very few special energy functions such as $f_e(x) = x$, the energy saving algorithms have the same performance as the shortest path strategy.

Third, the energy saving performance largely depends on the energy function. In our experiment, the convex energy function has a better energy saving efficiency than the linear energy function. And the different convex energy functions also do not have the same energy saving ability.

IV. CONCLUSION

In this paper, we consider a new time-slot scheduling algorithm in the network topology model considering the link capacity constraints. Our algorithm is a network-wide energy optimization method. By using this algorithm, we can deploy the traffic demands on the network within more than one time slots. We point out this idea because the one-time scheduling only gives a relaxed lower bound of the demanding routing process and causes the bandwidth waste, link idle and the unnecessary energy consumption especially under the capacity restriction. In order to find the optimal energy saving solution when allocating demands into the time slots, we formulate this problem into a new optimization problem which can be solved by the Lingo software. We conduct the experiments with a real ISP topology dataset and we find that the time-slot scheduling algorithm performs best compared to the shortest path algorithm and the one-time scheduling in energy saving. And the performance of scheduling methods differs in various energy functions.

There are also many problems not solved. In our further research the delay of the time slot scheduling algorithm need to be studied and the simulation experiment will be conducted to find the best delay and performance tradeoff solution. The study of how different energy functions determine energy consumption remains a challenging problem. Moreover, with the problem scale become large and the NLP problem will be very hard and the effective approximation algorithm to this time-slot method can be explored to guarantee the algorithm complexity.

V. ACKNOWLEDGMENT

We thank Dr. Lisa Zhang of Bell Labs for the helpful discussion and providing us the datasets information and some inspiring ideas. This work is supported by Major International Joint Research Program of National Natural Science Foundation of China under 61020106002.

VI. REFERENCES

[1] S. Antonakopoulos, S. Fortune and L. Zhang. Power-aware Routing with Rate-adaptive Network Elements. IEEE GreenCom, 2010.

[2] M. Andrews, A. Fernancez Anta, L, Zhang, and W, Zhao. Routing for Energy Minimization in the Speed Scaling Model. INFOCOM : 2435-2443.2010.

[3] P. Raghavan and C.D. Thompson. Randomized rounding: a techniquefor provably good algorithms and algorithmic proofs. Combinatorica,7:365–374,1991.

[4] Rocketfuel: an ISP topology mapping engine. http://www.cs.washington.edu/research/networking/rocketfuel/.

[5] Sergiu Nedevschi, Lucian Popa, Gianluca Iannaccone, Sylvia Ratnasamy, and David Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In Jon Crowcroft and Michael Dahlin, editors, NSDI, pages 323–336. USENIX Association, 2008.