Guided-Evolving:Convergence to Globally Optimal Load Balance by Distributed Computing using Local Information

Yongcai Wang
Institute for Theoretical Computer Science
FIT, 4-609, Tsinghua University
Beijing, China
wangyc@tsinghua.edu.cn

Yuexuan Wang
Institute for Theoretical Computer Science
FIT, 1-203, Tsinghua University
Beijing, China
wangyuexuan@tsinghua.edu.cn

Xiao Qi
Institute for Theoretical Computer Science
FIT, 4-609, Tsinghua University
Beijing, China
qix08@mails.tsinghua.edu.cn

ABSTRACT

In this demonstration, we present a guided-evolvement framework for load balancing in wireless multi-hop networks, which leads to stability and globally optimal load-balancing by solely distributed computing using local information. In wireless multi-hop networks, nodes conduct distributed computing and rely on message reception to get awareness of neighborhood information. Globally optimal performances are commonly hard to pursue due to instability or local optimum problems. We propose that the key for distributed load balancing to overcome premature and local optimum is seeking for a qualitative guidance to guide the distributed nodes to evolve towards convergence and global optimum. "Homo-level sensors should be equal in load" is devised as the qualitative guidance and a distributed transmission probability evolving framework is designed and developed. The multi-hop network's evolutional progress from arbitrary load distribution to optimal load balancing is demonstrated. The demo contains two versions, a web-based version for interactive demonstration of large-scale networks; and a sensor network hardware version to show how the framework works on the MAC and routing layers.

1. INTRODUCTION

With the ongoing trends of wireless network of extremely large scale and smarter nodes, it becomes more and more important, necessary and feasible to pursue globally optimal performances from distributed approach. It inspires this guided-evolving framework which executes the idea of "think globally, act locally". In this framework, we investigate how global load-balancing can be achieved by fully distributed traffic assignment algorithms, and devise that finding a qualitative guidance to lead the distributed nodes to optimal status is the key to guarantee network convergence to global optimum.

We consider multi-hop wireless networks for data collection, where wireless nodes with equal initial energy are randomly deployed in 2D space and form multi-level topology based on their hop-distances to the sink. Nodes in the network collect data periodically and transmit data hop-by-hop towards a remote sink. There is a "load accumulation" phenomenon in such networks, where children's loads will be forwarded to parents and increase the parents' loads. Load balancing is to control routing and traffic assignments to approach even load distribution for homo-level nodes, which is critical and crucial for multi-hop networks to be longevity and less congested. Because homo-level nodes have identical initial energy, if they are forced to consume energy at different rates due to load unbalance, some early dead nodes with the most descendants will bring the biggest damage to the network's functionalities.

However, load balancing is a very challenging problem in multi-hop data collection networks. Previous approaches conducted research from both centralized and distributed ways. The centralized algorithms mainly approached from load-balanced routing tree construction using whole information of the network. However, such approaches were proved NP-Complete by Buragohain[1] and Liang [2][3] independently. Distributed algorithms pursued homo-level load balancing by solely distributed computing and local information[5][4], which is more practical. However, instability and local optimum are the main challenges. To the best of our knowledge, existing distributed load balancing algorithms cannot provide guaranteed stability and optimality.

2. BASIC IDEA AND NOVELTY

We address load-balancing problem by distributed transmission probability evolvement based on the innovation of "a qualitative guidance". The constraints and goal for distributed load balancing problem are as following :

2.1 Constraints and Goal

The constraints are:

1. Computing using local information; any node adjusts transmission probability using only following information: 1) current loads of parent candidates ; 2) its own current load; and 3) the current transmission probabilities from it to its parent candidates.

2.*Limited communication range*, i.e., the number of parent candidates is limited, which are neighbors in up-streaming level and within communication range.

3. *Concurrency unawareness*, i.e., nodes are unknown of sibling's concurrent probability modifications.

4. *Limited knowledge,* i.e., nodes are unknown of other nodes' load information except the parent candidates.

The goal is "Optimal load balancing in all levels". This optimality goal is regarding the best probability assignments by all nodes in all levels, which maximizes the Chebysev Sum Equality based load balancing metric[6]. The optimal status is not always that all homo-level nodes have even loads. The even-load case is called "perfect load balancing", which is sometimes not reachable by solely transmission probability adjustment. The sufficient and necessary condition for converging to "perfect load balancing" is proved in [7], which needs joint adaptation of transmission probability and transmission power for some special topology networks.

With above constraints and goal, we investigated existing distributed load balancing algorithms. The phenomena at surface which prevents the convergence to global optimum are observed to be "load oscillation" and "local convergence" phenomena. The profound reasons are found as 1) nodes' unawareness to siblings' (sensors in the same level) concurrent operations, 2) lack of a guidance to inform nodes about the optimal status. So nodes overreact to local observations. Further, we found similar challenges exist in our daily life in society. In daily life, people make distributed decisions and commonly make bad choice due to limited knowledge. But due to social intelligence, knowledge-limited people can rely on a guidance from a knowledge-rich leader or a prophet to be informed what to do. The guidance is some rule designed from global view, which directs a right way for the distributed actions. This commonly leads to global optimum by fully distributed and independent actions. It is noticeable that the guidance must be qualitative without using any global computing or additional information, otherwise, the distributed computing problem will be changed by breaking the constraints.

Therefore, we devise that the key for distributed load balancing to overcome instability and local optimal solution is seeking for a qualitative guidance to guide the distributed evolving towards the global optimum. Based on this idea, a guided-evolving algorithm is proposed. The expected balanced load of the kth level (denoted by $E(L_k)$) works as the guidance for the children in level $k+1$ to update their transmission probabilities. In every time step, using $E(L_k)$ as the guidance, the children in level $k+1$ adjust its transmission probabilities to turn its parent candidates' loads towards $E(L_k)$. Particularly, for a node j in level $k+1$ and one of its parent candidates i in level k, j adjusts transmission probability to i by $P_{j,i}^{t+1} = P_{j,i}^{t}\frac{E(L_k)}{L_{k,i}^{t}}/M$, where $M = \sum_{m=1}^{n_j} P_{j,m}^{t}\frac{E(L_k)}{L_{k,m}^{t}}$ is a normalizer to keep the sum of the transmission probabilities from j equal to 1; $L_{k,i}^{t}$ is the load of node i at time t. By substituting M into $P_{j,i}^{t+1}$, we can see $E(L_k)$ is counteracted, and actually doesn't appear in the algorithm. So the value of $E(L_k)$ is actually not needed. Only a qualitative guidance that "The homo-level nodes should have equal load" is required. The fact is that the total loads and the number of nodes are fixed in level k, when such a qualitative guidance is reached by nodes in level k, the balanced load $E(L_k)$ is autonomously reached by the nodes. Using this guidance, we designed and developed the distributed transmission probability evolving framework and formally proved its convergence and global optimality [7].

To our best knowledge, this is the first time that the load-balancing problem is solved by an easily implemented distributed algorithm with guaranteed convergence and optimality. It can directly benefit many applications for its simplicity and fully distributed feature. It is noticeable that when the transmission probabilities reach stable, the network's topology is a weighted graph(not a tree; weights on the links are transmission probabilities), which is different from previous NP-complete tree based approach. Whether weighted graph approach is NP or not is still an open problem. But from our experiment results, we saw the convergence speed was quite fast[7]. Only dozens of periods are needed for networks of hundreds nodes to converge.

So particularly in this demo, we demonstrate the real-time guided-evolving process to visually present how the guaranteed stability and global optimum are reached gradually by nodes' local transmission probability evolvement.

3. DEMONSTRATION

We will take two versions of demos to present guided-evolving from different aspects.

3.1 Interactive Scalable Demo

The first demo is a web-based tool, having a GUI, which is used for interactive and scalable demonstration for guided-evolving. The advantages of a web-based demo are: users can interactively set the network size, the communication range of nodes, etc. , and can generate and see guided-evolving in random large-scale networks. The load balancing performances can be easily plotted, and can be easily compared with the performances of other algorithms. This web-based demo focuses more on conception than implementation details. It assumes ideal MAC protocol and routing protocol supporting to the data collection and the load-balancing algorithm.

To help user better understand the convergence process, guided-evolving is demonstrated in a stepwise manner. In each step, the loads of nodes, the transmission probabilities of links and the Chebyshev Sum Inequality based load balancing performance for a selected level will be rendered. So that, the guided-evolving process from arbitrary load distribution to optimal load balancing can be easily tracked. In addition, user can select other distributed algorithms to compare the load balancing performances, from which the "load oscillation" problem and "local optimum" problem can be intuitively rendered.

3.2 Hardware Demo

The second demo uses wireless sensor network hardware (off-shelf IRIS node developed by Xbow[8]) , which demonstrates guided-evolving's implementation and its performances in real wireless networks. The IRIS sensor has 2.4GHz radio, a 7M CPU, AD/DA components and is powered by two AA batteries. One of the IRIS node is sink and the others are data collectors.

The hardware demo is more complex than the web-based demo. The distributed load balancing algorithm is implemented on the wireless sensors in TinyOS environment based on the supports of a "stair scheduling" algorithm[7] in routing layer and a TDMA based MAC protocol. Global time synchronization is the foundation of above algorithms.

After deployment, the sensors will experience two phases.

1. Initializing phase, during which, 1) sensors carry out global time synchronization to prepare synchronized clocks for periodical data collection; 2) sensors are organized into leveled structure based on their hop-distances to the sink; 3) sensors survey neighbors and find the parent candidates.

2. Runtime phase, sensors turn to online guided-evolving. The nodes sleep for most of time to conserve energy, and wake up periodically to collect data and carry out distributed transmission probability computing.

The waking up times are scheduled based on the nodes' level index, called "stair scheduling" [7], as shown in Fig1. In stair scheduling, a node's active period is separated into "R-Slot", "T-Slot" and "Syn-Slot". The "T-slot" of nodes in level n is aligned to the "R-Slot" of the nodes in level $n-1$, and is aligned to the "Syn-Slot" of the nodes in level $n+1$. So that, in "R-Slot" a node receives data from its children; in "T-Slot" it transmits data to parents; in "Syn-Slot" the node overhears parents' broadcasting to do online time synchronization to maintain time synchrony. A slot is further composed by serval small slices. Transmissions of homo-level nodes are scheduled by TDMA to select different slice for collision avoidance.

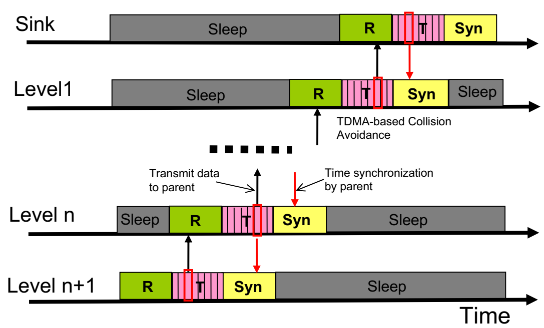

Figure 1: Stair scheduling to support level-based data aggregation and distributed load-balancing.

The work-flow of a node in an active period includes

In R-Slot, it collects local data; receives message from children and performs data aggregation.

In T-Slot, it reevaluates its current load based on the aggregation result, and recalculates the transmission probabilities to its parent candidates using the load balancing algorithm. Then it transmits the aggregated data to the parent candidates following the newly updated transmission probabilities.

In Syn-Slot, it hears broadcasting of parent nodes to do online time synchronization, and turns to sleep mode at the end of the active period.

In the hardware demo, for friendly rendering of guided-evolving, the load of sensors and the transmission probability information are added to the data message and are transmitted to the sink. The sink forwards the message to a laptop, and the laptop renders the real-time link weights and node loads on a graphical interface. The working status of sensors can also be seen from the three Led lights on the sensors.

4. REQUIREMENTS

We will bring a booth, two laptops and 10-20 IRIS nodes based on the availability of space. The booth is 60cm×160cm, which can be put besides a table. It is used to explain algorithm to audience if they are interested. The web-based interactive demo needs only a laptop with access to the internet. If one large LCD-screen (20" - 40") could be provided by organizers to show GUI, this would increase the user friendliness, otherwise we will use laptop screen. In the second demo, IRIS nodes will be deployed on a table. We will take out the antenna to form a multi-hop network while minimizing the required space. 20-40 AA batteries are needed to power up the sensors, but we think we can buy them from nearby convenience store. Another laptop is used in the second demo to show the network's topology, and real-time loads of the sensors. A power extension cord and two plug converters are needed for the two laptops. The total required area for the two demos is 1.5m×3m. The setup time of the demos is about two hours.

4.1 Other Info

This demo is eligible for the student demo competition. The lead student will be Xiao Qi. He is now a master student in Institute for Theoretical Computer Science, Tsinghua University. His email is qix08@mails.tsinghua.edu.cn. The practical development of the demonstration is based in large part his work.

5. ACKNOWLEDGEMENT

This research is supported by the National Basic Research Program of China Grant 2007CB807900, 2007CB807901, the National Science Foundation of China under grant No. 60604033, and the national 863 high tech R&D program of the Ministry of Science and Technology of China under grant No. 2006AA10Z216.

6. REFERENCES

[1] C. Buragohain and D. Agrawal. et.al. Power aware routing for sensor databases. In Proc. INFOCOM05, pages 1747-1757, 2005.

[2] W. Liang and Y. Liu. Online data gathering for maximizing network lifetime in sensor networks. IEEE Trans. Mobile Computing, 6:2-11, 2007.

[3] W. Liang and J. Luo. Prolonging network lifetime for data gathering in wireless sensor networks. submitted to IEEE Trans. Computers, 2009.

[4] J. Gao and L. Zhang. Load-balanced short-path routing in wireless networks. IEEE Trans. Parallel Distrib. Syst., 17(4):377-388, 2006.

[5] T.Yan and Y. Bi. et.al. Probability based dynamic load-balancing tree algorithm for wireless sensor networks. In ICCNMC 05, pages 682-691, 2005.

[6] P. Hsiao and A. Hwang. et. al. load-balancing routing for wireless access networks. In INFOCOM 01, 2001.

[7] Y. Wang. Optimal distributed load-balancing in multi-hop networks for data collection, technical report on tsinghua university. http://www.itcs.tsinghua.edu.cn/ yongcai/balancing.pdf, 2010.

[8] Crossbow introduces IRIS wireless product line, http://www.xbow.com, 2007