Proceedings of the 5th WSEAS Int. Conf. on Software Engineering, Parallel and Distributed Systems, Madrid, Spain, February 15-17, 2006 (pp163-168)

Load Distribution in Daisy Chain & Two-level d-dimensional Tree Networks Based on Power Consumption

SAMEER BATAINEH and MANAR ARAFAT
Computer Engineering Department
Jordan University of Science and Technology
P.O Box 3030 Irbid 12211
JORDAN

Abstract: - The ad hoc network is a self-organizing wireless system. Ad hoc networks today are playing an important role in some application environments where a decentralized network configuration is a functional advantage. This paper presents a mathematical model for load distribution in a distributed computing ad hoc network. The model captures various parameters which have significant impact on the node's battery life in order to achieve a balance in power consumption amongst the nodes after distribution and computing. This takes the form of closed form solutions for the optimal data allocation over processors interconnected in either a daisy chain or a two-level d-dimensional tree networks. The work presented is different than all previous work in that it considers the equilibrium of energy in all nodes as the fundamental element in distributing the load rather than the finish time. This is important because in an ad hoc network, each node is very crucial because it is considered as a link. So it is essential to have each node stay for the longest time in the system in order to improve the overall system performance.

Key-Words: - ad hoc network, divisible job, power consumption, tree network, load balancing, daisy chain.

1 Introduction

The ad hoc network is a self-organizing wireless system. This autonomous system is formed by a set of mobile hosts communicate through multi-hop wireless links without the aid of a central entity or any fixed infrastructure. Ad hoc networks today are playing an important role in some application environments where a decentralized network configuration is a functional advantage or even a necessity [1]. There are situations, where it is not feasible to have a wired network and ad hoc network is possibly the only solution. Examples include disaster recovery, battlefield communications, electronic classrooms, sharing information among participants at a conference, etc [2].

Ad hoc networks introduce a set of networking challenges differ from that of traditional wireless networks. Obviously, such networks are expected to operate under dynamic distributed environment; as a result, security and routing have become challenging tasks. Moreover, the network nodes will be battery-driven, each operating solely on its frugal energy budget. Hence, the energy resources are more limited in ad hoc wireless networks than in traditional wireless networks [3] [4].

This limitation in the nodes' battery capacity implies that the time during which each node functions properly is limited, which places a major constraint on the functional lifetime of the entire network. This is because losing a node power is like a node becomes faulty in wired system. This is in turn causes a communication cut off and if that node participates in executing parallel job then, a part of the job may not be completed.

Also some nodes are exposed to a faster depletion of their batteries than others; this mainly depends on the operations performed within the node. Generally, having most of the nodes alive for most of the time will improve the network stability considerably.

The ever increasing data-intensive tasks have created a need for dividing the data amongst multiple processors to improve the speed of computation through parallel computing. Parallel computing is the most effective way of exploiting the processing capability of the entire network in order to achieve faster solution time. As an evident, most of recent applications such as signal and image processing, experimental data processing, Kalman filtering, cryptography, and genetic algorithms, all involve parallel and distributed computing in order to improve system performance [1]. These applications lend themselves easily to the divisible load theory (DLT) [1] [2] [3] [6] [7] [8] [9]. Therefore we will deal with a class of divisible loads in our research as it covers many applications. All of the prior work on divisible load analysis involves an optimal allocation of load in order to minimize the finish (processing) time [1] [2] [3] [7] [9]. In our work we will distribute the load based on balancing energy consumption rather than equalizing the finish time of nodes. Therefore, divisible load theory (DLT) for ad hoc wireless networks must be mindful of the energy consumption besides the finish (processing) time. This is very important to consider in ad hoc networks since each node is very crucial because it is considered as a link. So it is essential to have each node stay for the longest time in the system. This urges the need of extending existing Load distribution strategies for ad hoc networks in new ways in order to meet the needs of applications besides regulating the energy consumption amongst the nodes. However, the inconveniency of nodes' battery replacement or recharging transforms energy conserving into one of the major challenges in ad hoc network design.

In this paper, we find closed form solutions for power equilibrium and the optimal load allocation over processors interconnected in either a daisy chain or a 2-level d-dimensional tree networks. The processors are all assumed to have the same packet transmission, reception and processing energy consumption. This makes it possible to find a closed form solution for the optimal load assignment to each processor in order to achieve power equilibrium after distribution.

This paper is organized as follows. In the second section, a daisy chain of processors is examined while in the third section we examine a 2-level d-dimensional tree network.

2 Daisy Chain Network

Consider a daisy chain of *n* processors. In a chain we have three types of nodes: root, intermediate and terminal nodes. The first node in the chain originates the load and becomes the root of the chain. The root processes a fraction of load itself while transferring the rest to the next node in the chain. If the next node is an intermediate node, it keeps a fraction of what it receives and transmits the remaining to the next node in the chain and so on till reaching the last node. The terminal node, which is the last node in the chain, does only receiving and processing.

It is assumed that the root node and each intermediate node will have a complete knowledge of the shortest multi-hop routes to its own children and the energy budget of each node in the distributed system.

Let us state the following notations which will be used thereafter:

b_i : The battery charge of node *i* (in joule).

E_i : The total energy consumed by node *i*.

L_i : The remaining energy in node *i*.

α_i : The fraction of load that is assigned to node *i* by the root node.

Et : The energy required by the root or any parent node to transmit the entire load to any node (in joule per load).

Er : The energy required by any node to receive the entire load (in joule per load).

Ep : The energy required by any node to process the entire load (in joule per load).

We assume a normalized load. Therefore, the fractions of the total measurement load should sum to one:

$$\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + ... + \alpha_n = 1 \quad (1)$$

The individual node energy consumption depends on the type of the node. Let us start with the 1st node and the *n*th node which are the root node and the terminal node respectively. The 1st node consumes energy while processing its load fraction and transferring the rest of the load to the next node in the chain. The total energy consumed by the root node can be given by the following equation:

$$E_1 = \alpha_1 Ep + (1-\alpha_1)Et \quad (2)$$

The *n*th node consumes energy while receiving and processing its load fraction. The energy consumed by the terminal node is given by:

$$E_n = \alpha_n (Ep + Er) \quad (3)$$

On the other hand, an intermediate node processes a fraction of what it receives, and transmits the remaining to the next node; the energy consumed by the intermediate nodes *2, 3, …, n-1* is given by:

$$E_2 = \alpha_2 Ep + (1-\alpha_1)E_r + (1-\alpha_1-\alpha_2)E_t \quad (4)$$

$$E_3 = \alpha_3 Ep + (1-\alpha_1-\alpha_2)E_r + (1-\alpha_1-\alpha_2-\alpha_3)E_t \quad (5)$$

.

.

$$E_{n-1} = \alpha_{n-1} Ep + \left(1-\sum_{i=1}^{n-2}\alpha_i\right)Er + \left(1-\sum_{i=1}^{n-1}\alpha_i\right)Et \quad (6)$$

In general, the energy consumed by intermediate node j is given by the following equation:

$$E_j = \alpha_j Ep + \left(1-\sum_{i=1}^{j-1}\alpha_i\right)Er + \left(1-\sum_{i=1}^{j}\alpha_i\right)Et$$

where

$$j = 2,..., n-1 \quad (7)$$

The remaining energy in the root node battery after processing its load fraction and transferring the rest of the load to the next processor is given by the following equation:

$$L_1 = b_1 - \alpha_1 Ep - (1-\alpha_1)Et \quad (8)$$

And the remaining energy in the terminal node battery after processing and receiving its underlying fraction is given by:

$$L_n = b_n - \alpha_n (Ep + Er) \quad (9)$$

For the intermediate processor j, the remaining energy after receiving, processing and transmitting is given by the following equation:

$$L_j = b_j - \alpha_j Ep - \left(1-\sum_{i=1}^{j-1}\alpha_i\right)Er - \left(1-\sum_{i=1}^{j}\alpha_i\right)Et$$

where

$$j = 2,..., n-1 \quad (10)$$

The objective in analyzing the above equations is to calculate the optimal load fractions to equalize the remaining batteries in all the processors after finishing the entire load:

$$L_1 = L_2 = L_3 = \dots\dots = L_n \quad (11)$$

Let us consider as an example a chain of four processors, n=4. The equations that govern the relation among various system parameters are:

$$L_1 = b_1 - \alpha_1 Ep - (1-\alpha_1)Et \quad (12)$$

$$L_2 = b_2 - \alpha_2 Ep - (1-\alpha_1)E_r - (1-\alpha_1-\alpha_2)E_t \quad (13)$$

$$L_3 = b_3 - \alpha_3 Ep - (1-\alpha_1-\alpha_2)E_r - (1-\alpha_1-\alpha_2-\alpha_3)E_t \quad (14)$$

$$L_4 = b_4 - \alpha_4(Ep + Er) \quad (15)$$

Solving recursively equations (1) and (12) to (15) based on equation (11), in our example:

$$L_1 = L_2 = L_3 = L_4$$

The system equations can be found by equating equations (12) and (13)

$$(b_1 - (b_2 - Er)) = \alpha_1(Ep + Er) + \alpha_2(Et - Ep)$$

By equating equations (13) and (14)

$$b_2 - b_3 = \alpha_2(Ep + Er) + \alpha_3(Et - Ep)$$

And by equating equations (14) and (15)

$$b_3 - b_4 = \alpha_3(Ep + Er) + \alpha_4(Et - Ep)$$

In order to simplify the derivation define:

$$M = Ep + Er$$

$$N = Et - Ep$$

$$K = Et + Er$$

One can rewrite the system equations as follows:

$$(b_1 - (b_2 - Er)) = \alpha_1 M + \alpha_2 N \quad (16)$$

$$b_2 - b_3 = \alpha_2 M + \alpha_3 N \quad (17)$$

$$b_3 - b_4 = \alpha_3 M + \alpha_4 N \quad (18)$$

Now, we have four equations (1), (16), (17), (18) with four variables α_1, α_2, α_3, α_4. Using (16) to find α_1 in terms of α_2:

$$\alpha_1 = \frac{(b_1 - (b_2 - Er))}{M} - \frac{N}{M}\alpha_2 \quad (19)$$

Similarly α_2 and α_3 are found in terms of α_3, α_4 respectively:

$$\alpha_2 = \frac{(b_2 - b_3)}{M} - \frac{N}{M}\alpha_3 \quad (20)$$

$$\alpha_3 = \frac{(b_3 - b_4)}{M} - \frac{N}{M}\alpha_4 \quad (21)$$

From equations (20) and (21) one can find α_2 in terms of α_4:

$$\alpha_2 = \frac{1}{M}\left[(b_2 - b_3) - \frac{N}{M}(b_3 - b_4)\right] + \left(\frac{N}{M}\right)^2 \alpha_4 \quad (22)$$

Similarly from equations (22) and (19) one can find α_1 in terms of α_4:

$$\alpha_1 = \frac{1}{M}\left[(b_1 - (b_2 - Er)) - \frac{N}{M}(b_2 - b_3) + \left(\frac{N}{M}\right)^2 (b_3 - b_4)\right] - \left(\frac{N}{M}\right)^3 \alpha_4 \quad (23)$$

Using equation (1) one can find α_4 in terms of system parameters as follows:

$$\alpha_4 = \frac{1 - \frac{1}{M}\left[((b_1 + Er) - b_4) - \frac{N}{M}(b_2 - b_4) + \left(\frac{N}{M}\right)^2 (b_3 - b_4)\right]}{\left[1 - \left(\frac{N}{M}\right) + \left(\frac{N}{M}\right)^2 - \left(\frac{N}{M}\right)^3\right]} \quad (24)$$

It follows from equations (21), (22), (23) and (24) that α_1, α_2, α_3, α_4 can be found in terms of various system parameters. In general, for a daisy chain of *n* processors, the root node load fraction in terms of the *n*th node load fraction is given by:

$$\alpha_1 = \frac{1}{M}\left[((b_1 + Er) - b_2) + \sum_{j=1}^{n-2}\left(\frac{-N}{M}\right)^j (b_{j+1} - b_{j+2})\right] + \left(\frac{-N}{M}\right)^{n-1} \alpha_n \quad (25)$$

The *ith* intermediate node load fraction in terms of the *n*th node load fraction is given by:

$$\alpha_i = \frac{1}{M}\left[\sum_{j=0}^{n-i-1}\left(\frac{-N}{M}\right)^j (b_{j+i} - b_{j+i+1})\right] + \left(\frac{-N}{M}\right)^{n-i} \alpha_n$$

where

$$i = 2, \dots, n-1 \quad (26)$$

And the terminal node (the *n*th node) load fraction in terms of system parameters is given by the following equation:

$$\alpha_n = \frac{1 - \frac{1}{M}\left[((b_1 + Er) - b_n) + \sum_{j=1}^{n-2}\left(\frac{-N}{M}\right)^j (b_{j+1} - b_n)\right]}{\sum_{j=0}^{n-1}\left(\frac{-N}{M}\right)^j} \quad (27)$$

3 Tree Network

Consider the 2-level 3-dimesional tree network of processors depicted in Fig.1. In the tree we have three types of nodes: root, intermediate and terminal nodes. The node that originates the load becomes the root of the tree. Each tree has only one root. An intermediate node can be viewed as a child of the root node. Also it is a parent of the terminal nodes. The terminal nodes can only be children nodes. It will be assumed that the root node and each intermediate node will have a complete knowledge of the shortest multi-hop routes to its own children and the energy budget of each node in the distributed system.

The individual processor energy consumption depends on the type of the node. The root node consumes energy while processing its load fraction and transferring the rest of the load to the intermediate processors in the first level. The total energy consumed by the root node (node 1 in Fig.1) can be given by the following equation:

$$E_1 = \alpha_1 Ep + (1 - \alpha_1) Et \tag{28}$$

Each intermediate node (nodes 2, 6, 10 in Fig.1) in the first level of the tree, consumes energy while receiving its own fraction and its children fractions of load, processing its fraction and transferring the rest to its terminal nodes for processing. In general the energy consumed by the *i*th node in the first level of a 2-level d-dimensional tree can be given by the following equation:

$$E_i = \alpha_i Ep + \left(\sum_{j=i}^{i+d} \alpha_j \right) Er + \left(\sum_{j=i+1}^{i+d} \alpha_j \right) Et \tag{29}$$

Each terminal node (nodes 3, 4, 5, 7, 8, 9, 11, 12, 13 in Fig.1) in the second level of the tree, consumes energy while receiving and processing its own load fraction. In general the energy consumed by the *i*th node in the second level of a 2-level d-dimensional tree can be given by the following equation:

$$E_i = \alpha_i (Er + Ep) \tag{30}$$

The remaining energy in the root node battery after processing its load fraction and transferring the rest of the load to the next processor is:

$$L_1 = b_1 - \alpha_1 Ep - (1 - \alpha_1) Et \tag{31}$$

In general, the remaining energy in the *i*th intermediate node battery in the first level of a 2-level d-dimensional tree can be given by the following equation:

$$L_i = b_i - \alpha_i Ep - \left(\sum_{j=i}^{i+d} \alpha_j \right) Er - \left(\sum_{j=i+1}^{i+d} \alpha_j \right) Et \tag{32}$$

And the remaining energy in the *i*th terminal node battery in the second level of a 2-level d-dimensional tree can be given by:

$$L_i = b_i - \alpha_i (Ep + Er) \tag{33}$$

One can rewrite the system equations (32) and (33) in terms of M, N and K as follows:

$$L_i = b_i - \alpha_i M - \left(\sum_{j=i+1}^{i+d} \alpha_j \right) K \tag{34}$$

$$L_i = b_i - \alpha_i M \tag{35}$$

The objective in analyzing the system equations (31), (34) and (35) is to calculate the optimal load fractions to equate the remaining batteries in all processor after finishing the entire load based on equation (11).
In general, in a 2-level d-dimensional tree of *n* nodes, the root load fraction in terms of the *n*th node load fraction will be:

$$\alpha_1 = \frac{((b_n + Et) - b_1)}{N} - \frac{M}{N} \alpha_n \tag{36}$$

And for each intermediate node in the first level of the 2-level d-dimensional tree, its load fraction in terms of the *n*th node load fraction is given by the following equation:

$$\alpha_i = \frac{1}{M}\left[(b_i - b_{i+1}) - \frac{N}{M}(b_{i+1} - b_n) - \frac{K}{M}\sum_{j=i+2}^{i+d}(b_j - b_n) \right] - \alpha_n \left[\frac{N + (d-1)K}{M} \right] \tag{37}$$

where
$i = 2,6,10$

Each child node in the second level of the tree, its load fraction in terms of the *n*th node load fraction will be:

$$\alpha_i = \frac{(b_i - b_n)}{M} + \alpha_n \tag{38}$$

where
$i = 3,4,5,7,8,9,11,12$

Finally the *n*th node load fraction in terms of system parameters is given by the following equation:

$$\alpha_n = \frac{1 + \frac{(b_1 - (b_n + Et))}{N} - \frac{1}{M}\left[\left[\sum_{i=2,6,10} b_i - d b_n \right] - \frac{N}{M}\left[\sum_{i=3,4,5,7,8,9,11,12} b_i - (d^2 - 1) b_n \right] \right]}{d^2 - \frac{M}{N} - d\left[\frac{N + (d-1)K}{M} \right]} \tag{39}$$

4 Conclusion and Future Work

Energy is crucial in ad hoc networks; each node has finite battery capacity and operates solely on its frugal energy budget. This limitation in the nodes' battery capacity implies that the time during which each node functions properly is limited, which places a major constraint on the functional lifetime of the entire network.

When the ad hoc network is used as a distributed computing environment, it becomes very essential to distribute the load such that the functional lifetime of the network is maximized, and the energy consumption at each node is regulated.

In this paper, we have obtained closed form solutions for power equilibrium and the optimal load allocation over processors interconnected in either a daisy chain or a two-level d-dimensional tree networks. The work presented is unique in that it considers the equilibrium of energy in all nodes as the fundamental element in distributing the load rather than the finish time.

This is very important to consider in ad hoc networks because losing a node power is like a node becomes faulty in wired system. Which in turn causes a communication cut off and if that node participates in executing parallel job then part of the job may not be completed. In general, keeping most of the nodes alive for most of the time will improve the network stability considerably, which improves the overall system performance.

Finally, we consider multi-level multi-dimensional tree network of processors in the context of load sharing as a future extension of this work.

References:

[1] Magnus Frodigh, Per Johansson and Peter Larsson, Wireless ad hoc networking—The art of networking without a network, Ericsson Review, Vol.77, No.4, 2000, pp. 248-263.

[2] M. Drozdowski and W. Glazek, Scheduling Divisible Loads in a Three-Dimensional Mesh of Processors, Parallel Computing, Vol.25, No.4, 1999, pp. 381-404.

[3] Mequanint A. Moges, Leonardo A. Ramirez, Carlos Gamboa and Thomas G. Robertazzi1, Monetary Cost and Energy Use Optimization in Divisible Load Processing, Proceedings of the 2004 Conference on Information Sciences and Systems, 2004 March, pp.17-19, Princeton, New Jersey.

[4] Carla-Fabiana Chiasserini, Imrich Chlamtac, Paolo Monti and Antonio Nucci, An Energy-Efficient Method for Nodes Assignment in Cluster-Based Ad Hoc Networks, Wireless Networks, Vol.10, No.3, 2004, pp. 223–231.

[5] E.Ilker Oyman and Cem Ersoy, The Effect of Overhead Energy to the lifetime in Wireless Sensor Networks, Proceedings of the 13th IST Mobile and Wireless Communications, 2004 June, pp. 27-30, Lyon, France.

[6] Laura Marie Feeney, Martin Nilsson, Investigating the Energy Consumption of a wireless Network Interface in ad hoc Networking. Proceedings of the IEEE INFOCOM, 2001 April, pp. 1548-1557, Paris, France.

[7] Sheetalkumar Doshi, Timothy X Brown, Minimum Energy Routing Schemes for a Wireless Ad Hoc Network, Proceedings of the IEEE INFOCOM, 2002 November, Paris, France.

[8] Suman Banerjee, Archan Misra, Adapting Transmission Power for Optimal Energy Reliable Multi-hop Wireless Communication, To appear in Wireless Optimization Workshop (WiOpt'03), 2003 March, Sophia-Antipolis, France.

[9] Black Box network services. (No date) 802.11: Wireless Networking. (Online) (Accessed 2005 August). Available from URL http://www.blackbox.com.

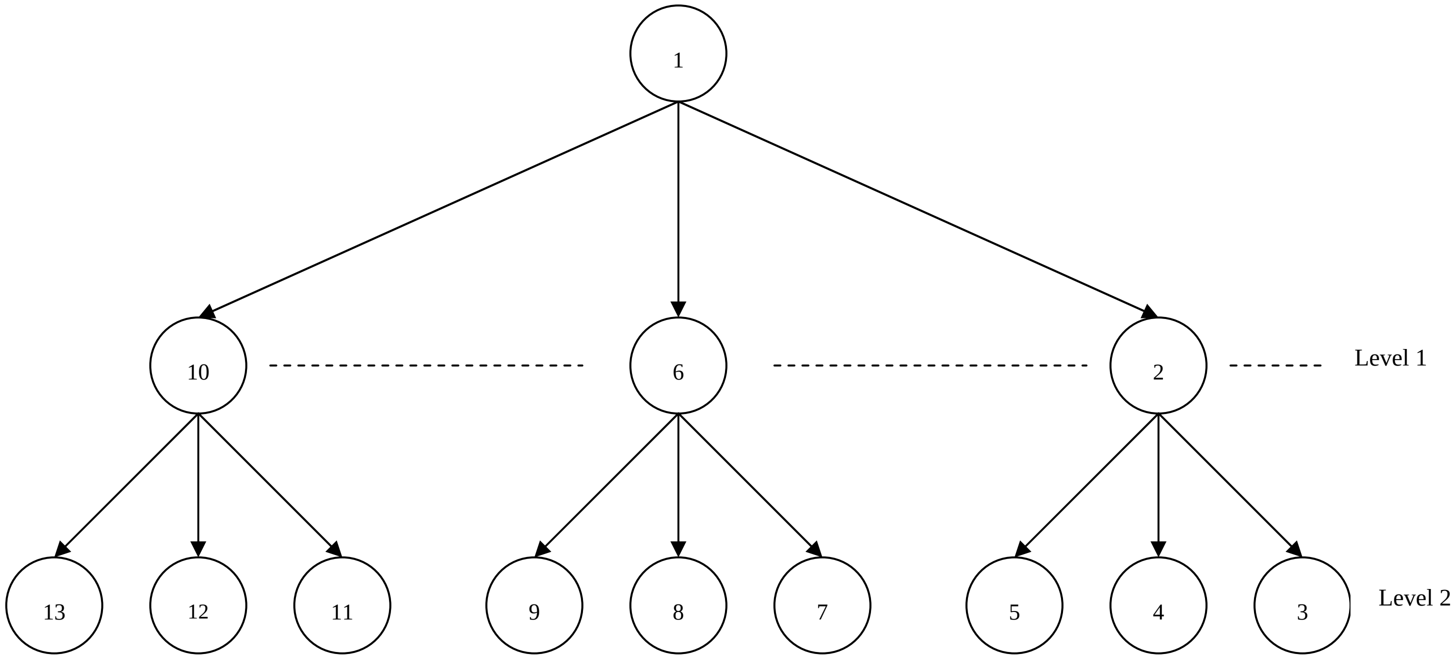

Fig. 1: 2-level 3-dimensional tree network