An Innovative Power-Saving Algorithm to Attain Extreme Throughput in Wireless Networks

Marada Tapaswi
M.Tech Scholar, Department of Computer Science and System Engineering,
Andhra University College of Engineering (A), Visakhapatnam, AP, India.

Abstract: In this paper, we consider the problem for maximizing the throughput of a discrete-time wireless network, where only certain sets of links can transmit simultaneously. It is well known that each set of such links can be represented by a configuration vector and the convex hull of the configuration vectors determines the capacity region of the wireless network. In the literature, packet scheduling polices that stabilize any admissible traffic in the capacity region are mostly related to the maximum weighted matching algorithm (MWM) that identifies the most suitable configuration vector in every time slot. Unlike the MWM algorithm, we propose a dynamic frame sizing (DFS) algorithm that also stabilizes any admissible traffic in the capacity region. The DFS algorithm, as an extension of our previous work for wired networks, also does not have a fixed frame size. To determine the frame size, an optimization problem needs to be solved at the beginning of each frame. Once the frame size is determined, a hierarchical smooth schedule is devised to determine both the schedule for configuration vectors and the schedule for multicast traffic flows in each link. Under the assumption of Bernoulli arrival processes with admissible rates, we show that the number of packets of each multicast traffic flow inside the wireless network is bounded above by a constant and thus one only requires implementing a finite internal buffer in each link in such a wireless network

***Index Terms: Wsn*, The DFS algorithm.**

INTRODUCTION

Packet scheduling in both wired and wireless networks to achieve maximum system throughput or provide quality of service has been an ongoing research problem for a long period of time. Our objective in this paper is to extend the dynamic frame sizing (DFS) algorithm for switches and wired networks to the setting of wireless networks. For this, we consider the configuration vector model that is often used in the literature to model the effect of link interference in a wireless network. A configuration vector is a vector of indicator variables that specifies a set of links which are allowed to transmit packets at the same time in a wireless network. The configuration vector model then characterizes a wireless network by a set of configuration vectors. Such a model is also known as the generalized constrained queuing model . For the configuration vector model, it is well known that the capacity region for a wireless network is the convex hull of the set of configuration vectors. There are plenty of studies in the literature that addressed the packet scheduling problem for maximizing the throughput in such a wireless network model. In particular, a scheduling algorithm is called throughput-optimal if it can stabilize the network for arrival traffic with rates falling within the capacity region. Most of throughput-optimal scheduling schemes are related to the maximum weighted matching (MWM) algorithm in, which identifies the most suitable configuration vector according to the queue length information available at each time slot.

Unlike the MWM algorithm, there is no need for the DFS algorithm to solve an optimization problem in every time slot. In the DFS algorithm, time is partitioned into frames, and an optimization problem is solved to determine the frame size at the beginning of each frame. For a wireless network that implements per flow queuing for each multicast flow, the frame size is chosen to be the minimum amount of time, known as the minimum clearance time, to clear the backlogs observed at the ingress queues at the beginning of the frame. By so doing, the backlogs at the ingress queues at the beginning of a frame is then bounded above by the arrivals during the previous frame, and a packet that arrives at an ingress queue in one frame will leave the ingress queue and enter the network in the next frame. Thus, as long as the expected size of each frame is finite, the expected backlog at each ingress queue remains finite. For packets that have departed from their ingress queues and entered the network, the DFS algorithm provides each flow in a frame a guaranteed rate that is proportional to the backlog observed at the ingress queue at the beginning of that frame. Such a guaranteed rate service then ensures the number of packets of each flow inside the network is bounded above by a finite constant. As a result, every internal buffer is finite and this mitigates the problem of implementing unlimited internal buffers as pointed out].The extension of the DFS algorithm to the configuration vector model is not as straightforward as one might expect. There are two technical difficulties that need to be conquered for such an extension. First, unlike the models for switches and wired networks, there is no explicit expression for the frame size in the configuration model as the minimum clearance time is now a solution of an optimization problem. Without an explicit expression for the frame size, it would be difficult to use the large deviation argument to prove the finiteness of the expected frame size. Second, packet scheduling in the configuration vector model is much more complicated than that for switches and wired networks. In the configuration vector model, there are two levels of scheduling:

(i) The upper level for scheduling configuration vectors and (ii) The lower level for scheduling flows in each link. Providing guaranteed rate services in such a two-level schedule for each frame poses a challenging problem.
Our contributions of this paper are mainly to solve these two technical problems. For the problem on the frame size, we derive an explicit upper bound for the minimum clearance time that can be used for the large deviation argument. Our idea for the upper bound is to compare the minimum clearance time with the time to drain the backlog at each link with the rate equal to its arrival rate (even though the actual arrival rate is not known). By so doing, we are able to establish the connection between the external arrival rates and the frame size so that the large deviation argument can be used. To provide guaranteed rate services inside the network, we propose a hierarchical smooth schedule (as an extension of the smooth schedule). We then derive bounds on the differences between the guaranteed rate services provided by the hierarchical smooth schedule and the ideal rate services. These bounds are then used for bounding the number of packets in each internal queue.

The rest of this paper is organized as follows. First, we describe our mathematical model in Section II. In Section III, we propose our DFS algorithm, including determining the frame size in Section III-A, proposing the hierarchical smooth schedule in Section III-B, and proving the finiteness of the expected frame size in Section III-C. In Section IV, we conclude this paper by addressing some further extensions.

DYNAMIC FRAME SIZING ALGORITHM

In [3], [8], the dynamic frame sizing (DFS) algorithm has been used for stabilizing queues in switches and wired net-works without knowing the arrival rates. In the DFS algorithm, time is partitioned into frames, where the frame size is not fixed and is determined at the beginning of each frame. The main idea of the DFS algorithm is to determine the minimum frame size at the beginning of a frame so that the backlog observed at the ingress queue of each flow at the beginning of the frame can be cleared by the end of the frame. To ensure the number of packets of each flow inside the network is bounded above by a finite constant, the DFS algorithm then provides each flow in a frame a guaranteed rate that is proportional to the backlog observed at the ingress queue at the beginning. As the set of configuration vectors is coordinate convex, one can always replace a configuration vector by a smaller configuration vector in the minimization problem in (7). As such, the minimization problem in (7) can be written equivalently (with an equality and possibly a larger set of configuration vectors) as follows:

EXISTING SYSTEM

Issues of power allocation and channel states (of transmitting links) were also taken into account. In particular, a Dynamic Routing and Power Control (DRPC) policy was proposed in to stabilize the queues by solving a joint routing and power allocation problem. Most of these works assumed infinite buffers. The effect of finite buffer-size on the performance of a network of queues was addressed. A standard approach to prove the stability of the MWM algorithm is to first consider a Lyapunov function and then show the existence of a negative drift of the Lyapunov function when the MWM algorithm (with back-pressure routing) is used. Another interesting approach is to use the carrier-sense multiple access (CSMA)-type random access algorithm to achieve the maximum throughput in ad hoc wireless networks. This approach usually requires a timescale separation assumption that assumes the CSMA Markov chain converges to its steady-state distribution instantaneously compared to the timescale of adaptation of the CSMA parameters.

PROPOSED SYSTEM

Motivated by the need for energy-efficient communication in wireless networks there has been tremendous interest in the study of various tradeoff mechanisms to achieve energy efficiency in each protocol layer. One of the main objectives of this paper is to study the delay-energy efficiency tradeoff in wireless networks by using the DFS algorithm. For this, we propose a new dynamic frame sizing algorithm, called the Greenput algorithm that takes power allocation into account. To obtain a good delay-energy efficiency tradeoff, the key insight of our Greenput algorithm is to reduce transmit power to save energy when the backlogs are low and this should not incur too much packet delay.

METHODOLOGY

L. Tassiulas and A. Ephremides, “Stability properties of constrained queuing systems and scheduling policies for maximum throughput in multi-hop radio networks,”
The stability of a queuing network with interdependent servers is considered. The dependency among the servers is described by the definition of their subsets that can be activated simultaneously. Multi-hop radio networks provide a motivation for the consideration of this system. The problem of scheduling the server activation under the constraints imposed by the dependency among servers is studied. The performance criterion of a scheduling policy is its throughput that is characterized by its stability region, that is, the set of vectors of arrival and service rates for which the system is stable. A policy is obtained which is optimal in the sense that its stability region is a superset of the stability region of every other scheduling policy, and this stability region is characterized. The behavior of the network is studied for arrival rates that lie outside the stability region. Implications of the results in certain types of concurrent database and parallel processing systems are discussed

N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, “Achieving 100% throughput in an input-queued switch,”
It is well known that head-of-line blocking limits the throughput of an input-queued switch with first-in-first-out (FIFO) queues. Under certain conditions, the throughput can be shown to be limited to approximately 58.6%. It is also known that if non-FIFO queuing policies are used, the throughput can be increased. However, it has not been previously shown that if a suitable queuing policy and scheduling algorithm are used, then it is possible to achieve 100% throughput for all independent arrival processes. In this paper we prove this to be the case using a simple linear programming argument and quadratic Lyapunov function. In particular, we assume that each input maintains a separate FIFO queue for each output and that the switch is scheduled using a maximum weight bipartite matching algorithm. We introduce two maximum weight matching algorithms: longest queue first (LQF) and oldest cell first (OCF). Both algorithms achieve 100% throughput for all independent arrival processes. LQF favors queues with larger occupancy, ensuring that larger queues will eventually be served. However, we find that LQF can lead to the permanent starvation of short queues. OCF overcomes this limitation by favoring cells with large waiting times.

J. G. Dai and B. Prabhakar, “The throughput of data switches with and without speedup,”
In this paper we use fluid model techniques to establish two results concerning the throughput of data switches. For an input-queued switch (with no speedup) we show that a maximum weight algorithm for connecting inputs and outputs delivers a throughput of 100%, and for combined input- and output-queued switches that run at a speedup of 2 we show that any maximal matching algorithm delivers a throughput of 100%. The only assumptions on the input traffic are that it satisfies the strong law of large numbers and that it does not oversubscribe any input or any output.

M. Armony and N. Bambos, “Queuing dynamics and maximal throughput scheduling in switched processing systems,”
We study a processing system comprised of parallel queues, whose individual service rates are specified by a global service mode (configuration). The issue is how to switch the system between various possible service modes, so as to maximize its throughput and maintain stability under the most workload-intensive input traffic traces (arrival processes). Stability preserves the job inflow–outflow balance at each queue on the traffic traces. Two key families of service policies are shown to maximize throughput, under the mild condition that traffic traces have long-term average workload rates. In the first family of cone policies, the service mode is chosen based on the system backlog state belonging to a corresponding cone. The issues of non-preemptive job processing and non-negligible switching times between service modes are addressed. The analysis is extended to cover feed-forward networks of such processing systems/nodes. The approach taken unifies and generalizes prior studies, by developing a general trace-based modeling framework (sample-path approach) for addressing the queuing stability problem. It treats the queuing structure as a deterministic dynamical system and analyzes directly its evolution trajectories. It does not require any probabilistic superstructure, which is typically used in previous approaches. Probability can be superposed later to address finer performance questions (e.g., delay). The throughput maximization problem is seen to be primarily of structural nature. The developed methodology appears to have broader applicability to other queuing systems.

A. L. Stolyar, “MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic,”
We consider a generalized switch model, which includes as special cases the model of multiuser data scheduling over a wireless medium, the input-queued cross-bar switch model and a discrete time version of a parallel server queuing system. The switch state follows a finite state, discrete time Markov chain. In each state m, the switch chooses a scheduling decision k from a finite set $K(m)$, which has the associated service rate vector. We consider a heavy traffic regime, and assume a Resource Pooling (RP) condition. Associated with this condition is a notion of workload is some fixed nonzero vector with nonnegative components. We study the MaxWeight discipline which always chooses a decision maximizing are arbitrary parameters. We prove that under MaxWeight scheduling and the RP condition, in the heavy traffic limit, the queue length process has properties, the workload process converges to a Reflected Brownian Motion. MaxWeight minimizes the workload among all disciplines.

P. Chaporkar and S. Sarkar, "Stable scheduling policies for maximizing throughput in generalized constrained queueing systems,"

We consider a class of queuing networks referred to as "generalized constrained queuing networks" which form the basis of several different communication networks and information systems. These networks consist of a collection of queues such that only certain sets of queues can be concurrently served. Whenever a queue is served, the system receives a certain reward. Different rewards are obtained for serving different queues, and furthermore, the reward obtained for serving a queue depends on the set of concurrently served queues. We demonstrate that the dependence of the rewards on the schedules alter fundamental relations between performance metrics like throughput and stability. Specifically, maximizing the throughput is no longer equivalent to maximizing the stability region; we therefore need to maximize one subject to certain constraints on the other. Since stability is critical for bounding packet delays and buffer overflow, we focus on maximizing the throughput subject to stabilizing the system. We design provably optimal scheduling strategies that attain this goal by scheduling the queues for service based on the queue lengths and the rewards provided by different selections. The proposed scheduling strategies are however computationally complex. We subsequently develop techniques to reduce the complexity and yet attain the same throughput and stability region. We demonstrate that our framework is general enough to accommodate random rewards and random scheduling constraints.

PROCESS

Network Model

This project assumes that N sensor nodes are randomly scattered in a two- dimensional square field A and the energy of sensor nodes cannot be recharged and each node in the network is assigned a unique ID. Also at any time, we assume that each sensor node is able to compute its distance to sink, its residual energy and its available buffer size (remaining memory space to cache the sensory data while it is waiting for servicing), as well as record the link performance between itself and its neighbours node in terms of signal-to noise ratio (SNR) and distance to sink.

Stability for Admissible Traffic

In this section, we show that for any admissible Poisson arrival traffic, the expected frame sizes under both the generic DFS algorithm in Algorithm 1 and the Greenput algorithm in Algorithm 3 are finite. Thus, all the queues are stable for such Poisson traffic. Here we make three specific assumptions on the input traffic. (A3) All the arrivals are independent Poisson processes. Specifically, the amount of data that arrives at link i during the time interval [s, t], denoted by Ai(s, t), is a Poisson random variable with mean λi(t − s). Without loss of generality, we assume that λi> 0 for all i = 1, 2, . . .,N. A4) Assume that the arrival rates λi, 1 ≤ i ≤ N, are unknown to the network.

Energy Efficiency

In our first simulation, we choose the minimum frame size for empty queues Tmin to be 1ms. we show the simulation results for energy efficiency (as a function of the threshold Tmax) under various traffic loads. Clearly, when Tmax is set to 0, the Greenput algorithm is always in the maximum power mode when there are backlogs. As such, all the data are transmitted with the maximum power allocation. This corresponds to the first data point. Thus the energy efficiency for Tmax = 0 is the worst and it is basically the same under various traffic loads, ranging from 0.1 to 0.9. As expected, increasing the threshold Tmax increases energy efficiency as it increases the chance for the algorithm to leave the maximum power mode and enter either the power -saving mode or the mixed power-saving mode. As such, more data are transmitted under TDMA schedules and that increases energy efficiency.

TDMA

In particular, the time division multiple access (TDMA) schedule (that has the least interference) is optimal under various channel assumptions. Inspired by these two important papers, our idea for a generic power-saving DFS algorithm is to use the minimum-time schedule with maximum power allocation in when the backlogs are large, and then switch to the minimum-energy schedule in when the backlogs are small. To determine whether the backlogs are small, we define a threshold Tmax (for the minimum time to empty the backlogs with maximum power allocation) and we only use the minimum-energy schedule (for saving energy) when the solution of the minimum-time schedule with maximum power allocation is smaller than the threshold Tmax.

CONCLUSION

In this paper, we extended the dynamic frame sizing algorithm to the setting of wireless networks. We modeled wireless networks by configuration vectors that specify the sets of links that can transmit at the same time. For such a mathematical model, we considered multicast flows with per flow queuing. We proved that the DFS algorithm indeed stabilizes the network for any admissible Bernoulli traffic. In comparison with the previous results for the DFS algorithm in [3], [8], our main contributions in this paper are the two new technical results: (i) an upper bound for the frame size that has an explicit expression in terms of the workload, and a hierarchical smooth schedule that provides guaranteed rate services in such a mathematical model. The first result allows us to modify the large deviation argument to prove the finiteness of the expected frame size, while the second result leads to an upper bound for the total number of packets in an internal queue.
There are some possible extensions of this work.

In (S1), the frame size is chosen to be 1 when there is no backlog in each ingress queue. In the worst case, this requires the optimization problem to be carried out in every time slot (which is as bad as the maximum weighted matching algorithm). We note that it is possible to set a bound on the minimum frame size so that the optimization problem needs not be carried out too often. This is because our proof only relies on the assumption that the backlog in each ingress queue at the beginning of a frame needs to be cleared by the end of that frame.

The DFS algorithm described in this paper does not provide traffic isolation. When the traffic is not admissible, the expected frame size cannot be bounded. As all the flows are coupled through the optimization problem that determines the frame size at the beginning of each frame, the performance could be very bad for all the flows. In view of this, one should enforce an upper limit for the frame size. But this also limits the throughput that can be achieved by the DFS algorithm.

REFERENCES

[1] C. S. Chang, Performance Guarantees in Communication Networks, London: Springer-Verlag, 2000.

[2] D. B. West, Introduction to Graph Theory, 2nd ed., Prentice-Hall, Inc. 2001.

[3] C. -S. Chang, Y. -H. Hsu, J. Cheng, and D. -S. Lee, “A Dynamic Frame Sizing Algorithm for CICQ Switches with 100% Throughput,” Proceedings of IEEE INFOCOM 2009.

[4] P. Chaporkar and S. Sarkar, “Stable scheduling policies for maximizing throughput in generalized constrained queueing networks, ”Proceedingsof IEEE INFOCOM 2006.

[5] P. Giaccone, E. Leonardi, and D. Shah, “Throughput region of finitebuffered networks, ”IEEE Transaction on parallel and distributedsystems, vol. 18, no. 2, Feb. 2007.

[6] S.-M. He, S.-T.Sun, H.-T.Guan, Q. Zheng, Y.-J. Zhao, and W.Gao,“On guaranteed smooth switching for buffered crossbar switches,”IEEE/ACM Transactions on Networking, vol. 16, no. 3, pp. 718-731,June 2008.

[7] L. B. Le, E. Modiano, and N. B. Shroff, “Optimal Control of Wireless Networks with Finite Buffers, ”Proceedings of IEEE INFOCOM 2010.

[8] C. M. Lien and C. S. Chang, “Generalized Dynamic Frame Sizing Algorithm for Finite-Internal-Buffered Networks, ”IEEE CommunicationLetters, vol. 13, no. 9, Sep. 2009.

[9] M. A. Marsan, A. Bianco, P. Giaccone, E. Leonardi, and F. Neri, “Multicast traffic in input-queued switches: optimal scheduling and maximumthroughput, ”IEEE/ACM Transactions on Networking, vol. 11, no. 3, June 2003.

[10] A. K. Parekh and R. G. Gallager, “A generalized processor sharing approach to flow control in integrated service networks: the multiplenode case,” IEEE/ACM Transactions on Networking, vol. 2, pp. 137- 150, 1994.

[11] H. Sariowan, R.L. Cruz and G.C. Polyzos, “Scheduling for Quality of Service Guarantees via Service Curves,” Proceedings of the InternationalConference on Computer Communications and Networks, 1995.

[12] G. Sharma, R. R. Mazumdar, and N. B. Shroff, “On the complexity of scheduling in wireless networks, ”MobiCom 2006, Sep. 2006.

[13] L. Tassiulas and A. Ephremides, “Stability properties of constrained queueing systems and scheduling policies for maximum throughput inmultihop radio networks,” IEEE Tra