Toward Formal Verification of 802.11 MAC Protocols: a Case Study of Applying Petri-nets to Modeling the 802.11 PCF

Russell Haines[*,†], Alistair Munro[*] and Gary Clemo[†]
[*] Centre for Communications Research, University of Bristol, UK
[†] Telecommunications Research Laboratory, Toshiba Research Europe Ltd., 32 Queen Square, Bristol, UK
Russell.J.Haines@bristol.ac.uk

Abstract — **Centralized control functions for the IEEE802.11 family of WLAN standards are vital for the distribution of traffic with stringent Quality of Service (QoS) requirements. These centralized control functions overlay a time-based organizational "super-frame" structure on the medium, allocating part of the super-frame to polling traffic and part to contending traffic. This allocation directly determines how well the two forms of traffic are supported. Given the vital role of this allocation in the success of a system, we must have confidence in the configuration used, beyond that provided by empirical simulation results. Formal mathematical methods are a means to conduct rigorous analysis that will permit us such confidence, and the Petri-net formalism offers an intuitive representation with formal semantics. We present an extended Petri-net model of the super-frame, and use this model to assess the performance of different super-frame configurations and the effects of different traffic patterns. We believe that using such a model to analyze performance in this manner is new in itself.**[1]

***Keywords** — WLAN; MAC; PCF; Petri-net; formal verification.*

I. Introduction

The vast majority of commercially available Wireless Local Area Networks (WLANs) employ the IEEE802.11 [1] family of standards, mainly 802.11b/g ("WiFi") to date. The current generation of WLAN deployments has been focused on wired LAN replacement for computer interconnection and data transfer, and, as such, predominantly supports only the contention-based access mechanism of IEEE802.11. However, WLANs are being increasingly presented as the technology of choice for distributing audio and video (A/V) content alongside more traditional data traffic.

Centralized control of the medium offers the ability to guarantee (as far as possible with a wireless medium) the delivery of A/V streams, whilst not starving the coexisting contending data traffic. Achieving and maintaining this balance requires sound and reliable solutions, which makes the mathematical rigor of formal methods appealing.

Formal methodologies permit the rigorous analysis of the correctness and stability of protocol and system models. Many such methodologies exist, from purely algebraic formulations to those with graphical elements, such as Petri-nets. The particular advantage of Petri-nets and their derivatives is that they form a link between the somewhat inaccessible algebras and more intuitive graphical models. Petri-nets are also particularly well suited to modeling concurrency and inter-process communication. This paper demonstrates that our model is relevant and realistic as a precursor to meaningful formal analysis.

II. IEEE802.11 Centralized Control

The original IEEE802.11 standard [2] included an optional centralized access mechanism (the Point Coordination Function, PCF), which has been extended within the IEEE802.11e standard [3] (as the Hybrid Coordination Function, HCF). The original PCF can be viewed as a constrained sub-set of HCF. A central coordinator (CC), co-located with the access point (AP), maintains a list of stations to be polled (determined when the stations *associate*) and polls them for traffic at pre-agreed intervals. The CC is able to seize the medium through prioritized access (achieved through the inter-frame spacing mechanism) and then orchestrate contention free traffic from the stations requiring it.

In order for this centralized scheme to coexist with the default contention-based access (the Distributed Coordination Function, DCF), a time-based super-frame structure is overlaid on the medium. This structure is marked by the transmission of a broadcast beacon, and then comprises a period of time during which contention-free polling takes place (the Contention Free Period, CFP), followed by a period of time during which contention-based access takes place (the Contention Period, CP)[2].

The time between successive beacon repetitions and the maximum proportion that can be spent in the CFP are set by the CC and broadcast to all stations via the beacon. These two parameters are referred to in the literature as CFP_{REP} and CFP_{MAX} respectively: this relationship is illustrated in Fig. 1. There are some restrictions: firstly, both the CFP and CP are not allowed to be smaller than one maximum-size exchange of each type, to guarantee that both types of traffic have a non-zero probability of gaining access. Secondly, if the central controller determines that no further polling is required in a given super-frame, it can end the CFP before the CFP_{MAX} limit is reached, donating the remainder to the CP.

[1] This work has been funded and supported by Toshiba Research Europe Limited: Telecommunications Research Laboratory.

[2] Note that, as part of the extensions of 802.11e, transmission opportunities can also be granted during the CP if so required.

0-7803-9392-9/06/$20.00 (c) 2006 IEEE

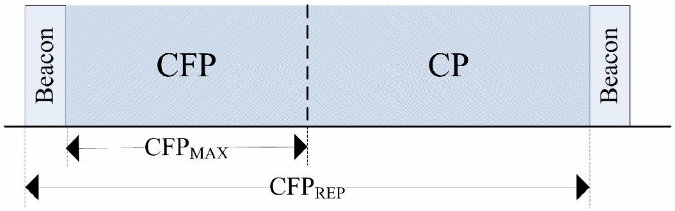


Figure 1: Super-frame Structure

III. Petri-nets and Reference-nets

Petri-nets were originally presented in Prof. Carl Adam Petri's landmark thesis of 1962 [4], and have spawned an entire research community that has been extending this work ever since. The original Petri-net formalism comprises *places*, *tokens*, *transitions* and *arcs*. Arcs join places and transitions. If tokens are present at a place this represents the availability of a resource; if all the places connected to a transition have sufficient tokens, then the transition may *fire* and deposit tokens in other places. The initial *marking* of places with tokens determines the initial conditions of the net. This seemingly simple behavior is representative of the most basic building blocks of communicating processes, and models of complex systems can quickly be built. A net can be expressed algebraically as a tuple representing the set of places, transitions, initial marking, and transfer functions, and can be manipulated mathematically, giving a robust and provable formalism [5, 6].

Extensions to these basic premises introduce rich functionality, including the extension of tokens to typed objects (as *colored nets* [7]), the ability to model hierarchies of functionality by referring to other nets as tokens [8] and the ability of transitions to invoke external actions. These higher order Petri-nets are well suited to modeling parallelism and communication between processes in a very compact manner, comparing favorably to the complexity of equivalent simulation models. Higher order Petri-nets have been successfully applied to modeling the IEEE802.11 DCF, the primary contention based access mechanism of the standard [9]. A variety of higher order Petri-net formalisms exist[3], and they lie on a broad spectrum of mathematical rigor and formalism, yet all have the fundamental basis of Petri-nets at the core.

One particular higher-level formalism is the Reference-net [12], which includes the ideas of colored nets and allowing tokens to be references to other nets, along with dynamic creation of net instances, synchronous communication between them, and the idea of *timed nets* where arcs can have durations associated with them. In the case of an arc leading from a place to a transition, the duration signifies that the tokens must remain available at the place for that period of time; in the case of an arc leading from a transition to a place, the token will not be deposited in the place until after that time.

Reference-nets are supported by the Reference-net Workshop (RENEW) tool [13][4], a Java-based tool that provides authoring, syntax checking and simulation tools for Reference-nets. RENEW's inscription language is Java and the importing of standard Java classes is supported. As a result, Java methods for deriving random numbers and employing distributions (such as the negative exponential for packet arrivals) are available to be used when determining durations in the timed net. The disadvantages of the loss of mathematical purity incurred by the inclusion of nearly the entire Java class library are offset by the added expressive power that results.

IV. This Model

This paper presents a Reference-net model of the IEEE802.11 centralized access control's super-frame. The medium is abstracted as an error- and collision-free channel with no hidden or exposed nodes. The model can be marked to represent different physical layer (PHY) characteristics. Two types of traffic are modeled, namely polled traffic with Quality of Service (QoS) requirements and contending traffic with no QoS requirements. Traffic generators for both types of traffic can be configured in a variety of ways, with any number of traffic generators of each type supported. The traffic generators are Reference-nets, instantiated by the parent net "pcf".

The contention-based data traffic generator (see Fig. 2, simplified with the background file-handling removed for clarity) generates packets at an arrival rate governed by a negative exponential distribution, the parameters of which are passed to it during initialization (and which are specified during the initial marking of the parent net). The size of the payload (or MAC Service Data Unit, MSDU) is governed by a truncated negative exponential distribution with an upper limit set as the IEEE802.11 maximum packet size and the mean packet size, again, configured during initialization.

The polled (QoS-sensitive) traffic generator (see Fig. 3, simplified again) is modeled as a simple on/off generator, with the on and off periods set on initialization. Whilst in the "on" state, the traffic generator emits packets of a fixed (configurable) size at a fixed (configurable) data rate. An additional feature of the polled traffic generator supports the QoS requirement of a *delivery deadline*. When a polled (QoS-sensitive) packet is generated, the generator uses a timed transition to generate a second "expiry" message after the maximum tolerable delivery delay for this stream has passed, which is delivered to the parent net.

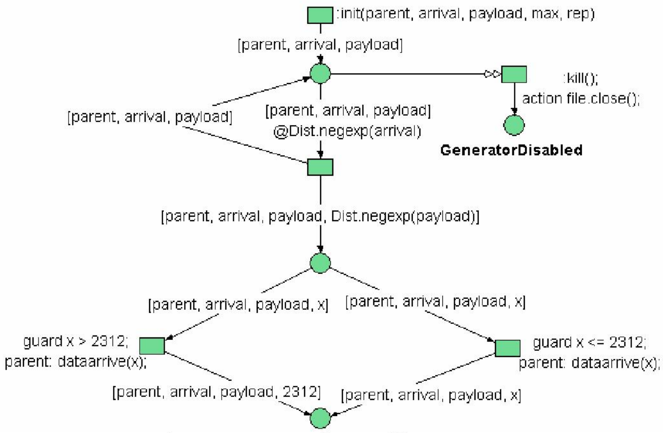

:init(parent, arrival, payload, max, rep)
[parent, arrival, payload]
:kill();
action file.close();
[parent, arrival, payload]
[parent, arrival, payload]
@Dist.negexp(arrival)
GeneratorDisabled
[parent, arrival, payload, Dist.negexp(payload)]
[parent, arrival, payload, x]
[parent, arrival, payload, x]
guard x > 2312;
parent: dataarrive(x);
guard x <= 2312;
parent: dataarrive(x);
[parent, arrival, payload, 2312]
[parent, arrival, payload, x]

Figure 2: Contending traffic generator

[3] E.g. Stochastic Petri-nets [9], [10] and PEPA-nets [11].

[4] See http://www.renew.de

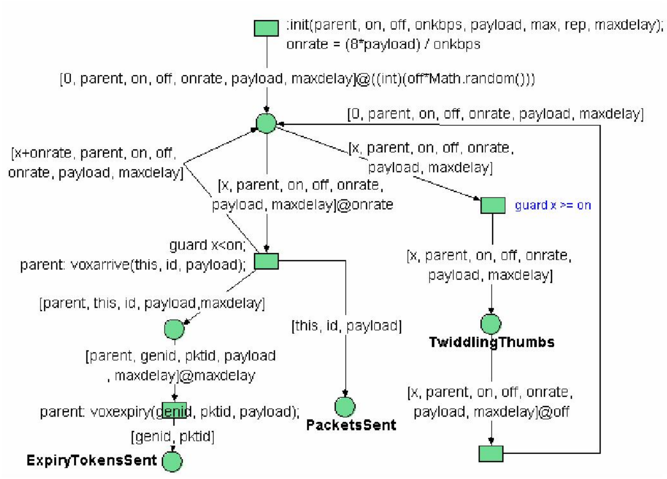
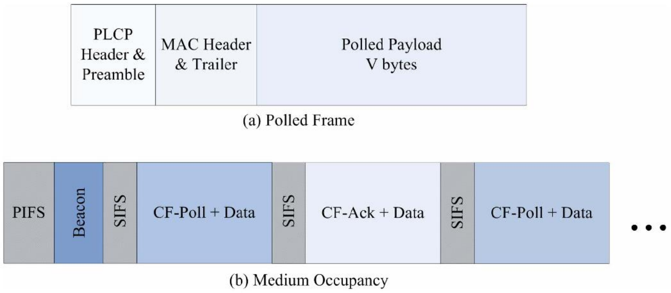

:init(parent, on, off, onkbps, payload, max, rep, maxdelay);
onrate = (8*payload) / onkbps
[0, parent, on, off, onrate, payload, maxdelay]@((int)(off*Math.random()))
[0, parent, on, off, onrate, payload, maxdelay]
[x+onrate, parent, on, off, onrate, payload, maxdelay]
[x, parent, on, off, onrate, payload, maxdelay]
[x, parent, on, off, onrate, payload, maxdelay]@onrate
guard x >= on
guard x<on;
parent: voxarrive(this, id, payload);
[x, parent, on, off, onrate, payload, maxdelay]
[parent, this, id, payload,maxdelay]
[this, id, payload]
TwiddlingThumbs
[parent, genid, pktid, payload , maxdelay]@maxdelay
parent: voxexpiry(genid, pktid, payload);
PacketsSent
[x, parent, on, off, onrate, payload, maxdelay]@off
[genid, pktid]
ExpiryTokensSent

Figure 3: Polled traffic generator

The main net "pcf" models the passing of time on the medium for this WLAN, with every packet transmission and inter-packet medium-access spacing (e.g. inter-frame spaces and back-off periods) modeled as duration. Having "sent" a beacon along with the associated spacings, the net polls all stations for which it has polling traffic; in this way it represents an infrastructure-based traffic source, as could be found in a wireless telephony system or, at higher data rates, the hub of a home entertainment system. The existence of polling traffic for a station is as a result of one of the aforementioned polled traffic generators having emitted a packet (MSDU); each MSDU is then appropriately wrapped with MAC and PHY overheads to give an overall air-time duration, as illustrated in Fig. 4. An assumption is that polled stations always have a packet to transmit; extension to the net to incorporate wasted polls is straightforward. Another extension is to consider polls that initiate point-to-point transmissions (rather than via the access point), and this is considered later in this paper.

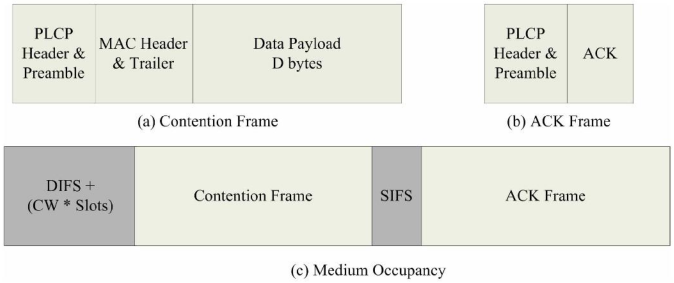
As soon as the central controller has run out of stations to poll, or CFP_{MAX} has been reached, the super-frame enters the CP. During the CP, packets generated by the contending data traffic generators are again wrapped with MAC and PHY overheads, including a representation of the IEEE802.11 random back-off that uses a random number of slots scaled to the minimum contention window size for the PHY in question (dictated by the initial marking), as illustrated in Fig. 5. Once the CFP_{REP} duration has expired, the net reiterates to the beacon stage for a new super-frame.



(a) Polled Frame

(b) Medium Occupancy

Figure 4: Polled frame structure



(a) Contention Frame

(b) ACK Frame

(c) Medium Occupancy

Figure 5: Contending frame structure

The delivery-deadline QoS requirement is handled by receiving the expiry message from the polled traffic generators, then matching the expiry message with the pending polled traffic packets, this being possible as each packet is uniquely identified. Expired packets are summarily removed and not sent, which penalizes the expired packet, but also improves the delay imposed on the remaining (unexpired) packets by improving the likelihood of their being sent before their deadlines expire.

The parent net "pcf" (Fig. 6, shown simplified for clarity) also has responsibilities for initialization, shutdown and file management. It creates the traffic generator sub-nets from the information provided in the initial marking, which also establishes the super-frame structure (from CFP_{MAX} and CFP_{REP}). The simulation is run for a predetermined time (specified as a number of beacons), with the passage of the superframe-state token down the left hand side (Beacon, CFP, CFPmax Reached, CP and CP-END places) marking the superframe sequence on the medium.

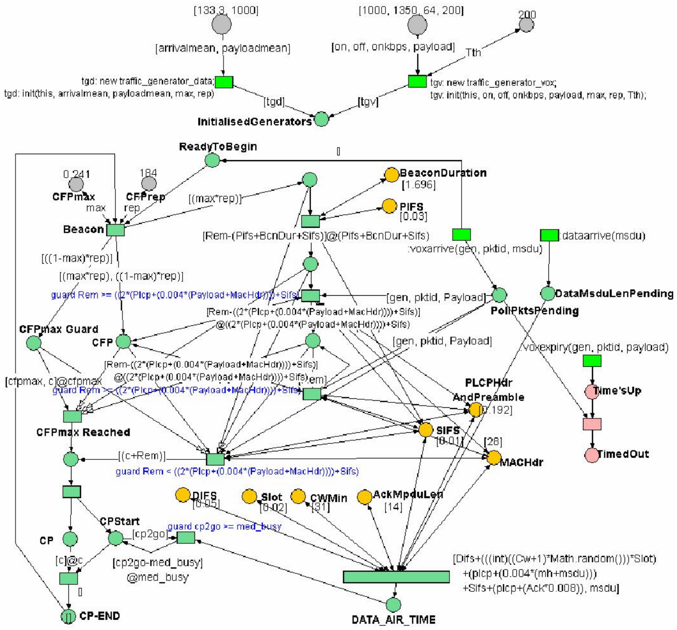

[133.3, 1000]
[1000, 1350, 64, 200]
200
[arrivalmean, payloadmean]
[on, off, onkbps, payload]
Tth
tgd: new traffic_generator_data;
tgd: init(this, arrivalmean, payloadmean, max, rep)
tgv: new traffic_generator_vox;
tgv: init(this, on, off, onkbps, payload, max, rep, Tth);
[tgd]
[tgv]
InitialisedGenerators
ReadyToBegin
0.241
184
CFPmax
CFPrep
max
rep
[(max*rep)]
Beacon
BeaconDuration
[1.696]
PIFS
[0.03]
[Rem-(Pifs+BcnDur+Sifs)]@(Pifs+BcnDur+Sifs)
:voxarrive(gen, pktid, msdu)
dataarrive(msdu)
[((1-max)*rep)]
[(max*rep), ((1-max)*rep)]
guard Rem >= ((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)
[gen, pktid, Payload]
PollPktsPending
DataMsduLenPending
[Rem-((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)]
@((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)
CFPmax Guard
CFP
[gen, pktid, Payload]
voxexpiry(gen, pktid, payload)
[Rem-((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)]
@((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)
[cfpmax, c]@cfpmax
guard Rem >= ((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)
[Rem]
PLCPHdr AndPreamble
[0.192]
Time'sUp
SIFS
[0.01]
[28]
CFPmax Reached
[(c+Rem)]
guard Rem < ((2*(Plcp+(0.004*(Payload+MacHdr)))+Sifs)
MACHdr
TimedOut
DIFS
[0.05]
Slot
[0.02]
CWMin
[31]
AckMpduLen
[14]
CP
CPStart
[cp2go]
guard cp2go >= med_busy
[c]@c
[cp2go-med_busy]
@med_busy
[Difs+(((int)((Cw+1)*Math.random()))*Slot)
+(plcp+(0.004*(mh+msdu)))
+Sifs+(plcp+(Ack*0.008)), msdu]
[]
CP-END
DATA_AIR_TIME

Figure 6: The net "pcf"

V. Results

To validate the model, CFP_{REP}/CFP_{MAX} combinations presented in a table-based look-up scheme for super-frame configuration by *Li et al.* [14-16] have been compared with results obtained from this model, with this model configured to the same PHY and traffic configurations; for each CFP_{REP} configuration modeled, the CFP_{MAX} value for acceptable delivery of all traffic is in the same region in both cases. For example, taking a sample of 200ms delay threshold values (note that the traffic generated by smaller numbers of terminals can be satisfied by a range of possible CFP_{MAX} values without adversely affecting the contending traffic):

Table 1: Example Comparison Points

Number of Polling STA	CFP_{REP}	Li's CFP_{MAX}	Model CFP_{MAX}
2	28ms	0.2	0.2-0.4
10	55ms	0.5	0.6
20	75ms	0.9	0.85

A shortcoming with Li's work [14-16] is that the values offered do not take into account the minimum CFP and CP sizes mandated by the standard, and all traffic flows go directly between stations rather than via the access point as would be the case in infrastructure deployments. Nonetheless, the correlation between Li's findings and those from this model give confidence in this model.

It is interesting to compare different polling traffic scenarios with this model, and the changes are readily implemented in the net. Three interesting configurations are presented here: an infrastructure-network traffic pattern with all data going via the access point (as may well be the case with a home entertainment network) with polled traffic allowed to accumulate irrespective of delivery deadlines; the same traffic model but with expired pending packets removed if not sent in time; and finally a station-to-station model like that of Li's, which models the Direct Link Protocol (DLP) of IEEE802.11e.

Taking, for example, the scenario of polled traffic with a delivery-delay threshold of 200ms again, for configurations of two, ten and twenty polled stations, the results for three different configurations are presented in Fig. 7, Fig. 8 and Fig. 9 below. In each figure there are three pairs of traces (one for the *basic* configuration, one for *exp*ired polled-packet removal, and one for the point-to-point *DLP* configuration), each pair comprising a contending (data) traffic throughput trace and a polled (voice) traffic throughput trace in each case.

The "Basic" and "Exp" traces show that the CFP_{MAX} required for full delivery is increased when frames are removed if their delivery deadline expires. Without the expiry mechanism polled traffic can simply accumulate and then be serviced during lulls in traffic generation (recall that the traffic generators are on/off models, albeit with staggered start times). With the expiry mechanism enabled, accumulated packets may well expire before they are scheduled to be delivered, resulting in an aggregate drop in packets delivered. Hence, to support the same level of throughput, the CFP_{MAX} size must be greater to cope with the peak demand within the tolerable delivery delay threshold specified.

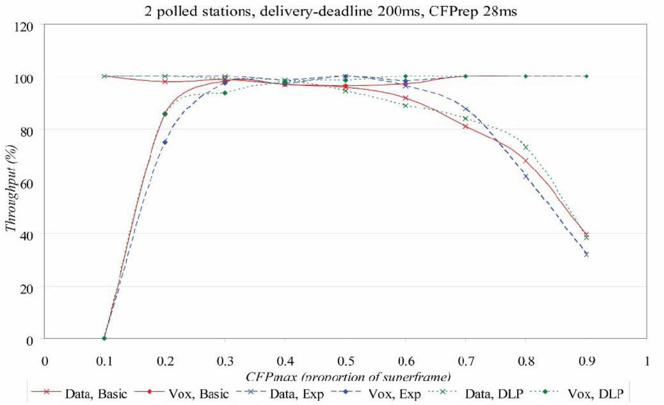


Figure 7: Two polled stations, maximum delay of 200ms

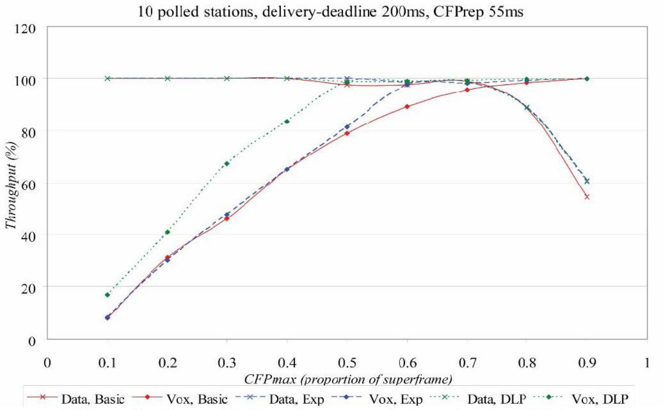


Figure 8: Ten polled stations, maximum delay of 200ms

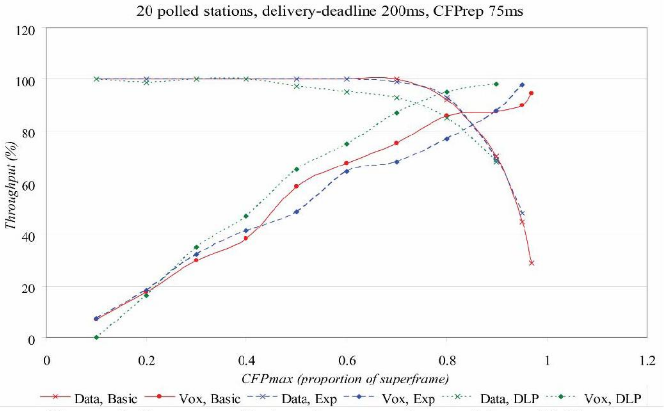


Figure 9: Twenty polled stations, maximum delay of 200ms

The effect of changing the traffic pattern from "via access point" to "station to station" is shown in the "Exp" and "DLP" traces respectively in the figures above. It can be seen that, as would be intuitively expected, reducing the bandwidth requirements of the polled traffic has reduced the demand for the CFP and hence decreased the required CFP_{MAX}. Note, however, that despite a 50% reduction in the MSDU/payload contribution to the required bandwidth, the resulting reduction in CFP requirements has dropped by far less as a result of the fixed overheads of the CF-Poll mechanism.

Overall, a clear pattern can be seen for each configuration. The fewer polled-traffic stations there are, and the smaller the bandwidth required by each, the more easily the requirements of both polled- and contending-traffic are met, resulting in a longer “sweet spot” where all flows are being serviced to a level within the required tolerances. As the polled traffic bandwidth increases, larger and larger CFP_{MAX} values are required until the contending traffic cannot be fully supported even at the most lavish of CFP sizes.

Hence, the total bandwidth of the polled traffic (i.e., a function of number of terminals and the bandwidth requirements of each) appears to be the primary determining factor in selecting CFP_{MAX}, regardless of the duration of $CFP_{REP.}$

VI. Conclusions

In this paper we have presented a model of 802.11 centralized polling applicable to both the PCF and HCF modes of operation, using the mathematically sound, yet graphical, formalism of Petri-nets.

We have expressed the super-frame in terms of transmission duration on the medium, modeling data traffic as arriving according to a negative-exponential distribution, with payload lengths determined by a truncated negative-exponential, and modeling polled traffic with on/off traffic generators employing a fixed data rate when on and a fixed packet size.

We applied our Reference-net model to the analysis of system performance in a way that has not been presented in the literature to date. After verifying that the results obtained from this model are consistent with published results on the same topic, we have shown the effects of implementing a “packet expiry” mechanism to police pending traffic and remove irrelevant packets, and some evidence toward the benefits of adopting the IEEE802.11e Direct Link Protocol to send packets directly to other stations within the same network rather than having to route via the access point. Various extensions of the model are possible, including the addition of different traffic generator types such as video traffic models for higher data-rate PHY configurations.

The value of this model is two-fold. Firstly, as a graphical model of the centralized polling aspects of the IEEE802.11 WLAN, it provides a compact and readily accessible representation of the system that is capable of generating basic simulation results for a wide variety of scenarios, traffic patterns and system parameters. The second and most important strength of this model is its potential for formal verification. Hence, from this graphical representation, the next stage in this work is to explore the mathematical aspects of this representation thoroughly, and exploit the capacity to prove the correctness of the model. The benefit of this work is that ensuing analysis will begin from a model that has been shown to be relevant and applicable to real-world problems.

References

[1] IEEE (Institute of Electrical and Electronics Engineers),"IEEE Wireless LAN Edition - A compilation based on IEEE Std 802.11TM-1999 (R2003) and its amendments," 2003.

[2] IEEE (Institute of Electrical and Electronics Engineers),"IEEE Std 802.11: Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[3] IEEE (Institute of Electrical and Electronics Engineers),"IEEE Std 802.11e - IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," 2005.

[4] C. A. Petri, "Kommunikation mit Automaten (English translation: Technical Report RADC-TR-65-377)," *PhD Thesis, University of Bonn (Eng: Vol.1, Suppl 1, Applied Data Research, Princeton, N.J.)*, 1962.

[5] G. Berthelot and R. Terrat, "Petri Nets Theory for the Correctness of Protocols," *IEEE Transactions on Communications*, vol. 30, iss. 12, pp. 2497 - 2505, 1982.

[6] N. Chamas and H. Singh, "A Generalized Petri Net State Equation," *Proceedings of the 32nd Midwest Symposium on Circuits and Systems, 1989.*, vol. 1, pp. 161 - 164, 1989.

[7] K. Jensen, *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use.*, vol. 1 (Basic Concepts): Springer-Verlag GmbH, 1997.

[8] R. Valk, "Petri Nets as Token Objects - An Introduction of Elementary Object Nets," *Lecture Notes in Computer Science: 19th Inernational Conference on the Application and Theory of Petri Nets (ICATPN '98)*, (Lisbon, Portugal), 1998.

[9] A. Heindl and R. German, "Performance modeling of IEEE 802.11 wireless LANs with stochastic Petri nets," in *Performance Evaluation*, vol. 44: Elsevier Science B.V., 2001, pp. 139-164.

[10] S. Donatelli, M. Ribaudo, and J. Hillston, "A comparison of performance evaluation process algebra and generalized stochastic Petri nets," *Proceedings of the Sixth International Workshop on Petri Nets and Performance Models*, pp. 158-168, 1995.

[11] S. Gilmore, J. Hillston, L. Kloul, and M. Ribaudo, "PEPA nets: A structured performance modelling formalism," *Computer Performance Evaluation, Modelling Techniques and Tools. 12th International Conference, TOOLS 2002*, pp. 111- 130, 2002.

[12] O. Kummer, "Introduction to Petri Nets and Reference Nets," *Sozionik Aktuell*, vol. 1, pp. 1-9, 2001.

[13] O. Kummer, F. Wienberg, and M. Duvigneau (University of Hamburg, Department for Informatics, Theoretical Foundations Group, Distributed Systems Group), "Renew - User Guide," *unpublished*, 2004.

[14] C. Li, J. Li, and X. Cai, "Self-adaptive transmission scheme of integrated services over an IEEE 802.11 WLAN," *Electronics Letters*, vol. 40, iss. 25, pp. 1596, 2004.

[15] C. Li, J. Li, and X. Cai, "Performance Analysis of IEEE 802.11 WLAN to Support Voice Services," *Proceedings of the 18th International Conference on Advanced Information Networking and Application (AINA '04)*, vol. 2, pp. 343 - 346, 2004.

[16] C. Li, M. Li, and X. Cai, "A study of self-adaptive transmission for integrated voice and data services over an IEEE 802.11 WLAN," *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004.*, vol. 3, pp. 1922 - 1926, 2004.