Modeling and Analysis of TCP in Wireless Networks using a Max-Plus Model

Rodrigo Andreola, Ricardo Lüders, Keiko V. Fonseca

Abstract—**Analytical models for packet networks have been widely proposed in the literature. Most of them are based on stochastic approaches which yields to results based on average values of probability distributions. However, a different approach is used in this paper: a discrete event system approach based on a deterministic max-plus model found in the literature. This model characterizes temporal behaviors of a flow of packets transmitted over TCP in a network composed by routers in a tandem configuration. Although a cabled network was used in the literature, this paper presents a performance analysis for the max-plus model in a wireless network. It is well known that TCP protocol performance is quite affected in wireless networks as discussed in the paper. By using the max-plus model, transmission rates are calculated and compared to those obtained by simulation. In addition, important issues for tuning parameters of the max-plus model are identified.**

Index Terms—**Max-plus algebra, network performance analysis, TCP protocol, wireless network.**

I. INTRODUCTION

Some applications executing over computer networks require a minimum quality of service (QoS) specified by pre-determined parameters and set by a service level agreement (SLA). These applications require models for verifying the efficiency of admission control techniques, and performance evaluation according to a SLA among others. Stochastic models have been widely chosen for these purposes. But it is not always possible to use analytical models without imposing simplification. As an alternative, discrete event simulation models for packet level are used to analyze flows and network performance. However, the complexity of these models is proportional to the huge amount of packets that have to be simulated to achieve reliable values. Therefore, its use is quite questionable for huge networks. Moreover, a broad application of wireless communication networks has yielded to a more complex analysis task.

Computer networks can also be viewed as a Discrete Event System (DES) with packet arrivals and departures (events) being suited modeled by different discrete event approaches. Among them, discrete event systems subject to only synchronization phenomena have been successfully modeled by max-plus algebra and similar ones [1]. Network Calculus (NC) is an important example of computer network modeling by using this algebraic approach [2] and [3]. NC is a set of techniques based on min-plus algebra which provides (worst case) bounds for network traffic. These bounds can be used to allocate resources and estimate network performance through algebraic operations.

In particular, many applications in the Internet depend on a reliable data transport which is commonly done by TCP (Transmission Control Protocol) [4]. TCP protocol provides reliable end-to-end transport and it is widely adopted as transport protocol for web applications. The protocol detects congestion events by controlling ordering and arrival time of acknowledgment messages and it sets a flow transmission rate to provide fairness on the network resource sharing. Due to its importance, different analysis has been proposed for studying its adaptive behavior for flow control [5], [6], and [7]. Recently, [8] has reported both complexity and time execution gains for the analysis of TCP behavior using models based on Network Calculus compared to ones obtained by discrete event and fluid models. A model for TCP using max-plus algebra proposed by [9] describes the behavior of a packet flow transmitted over TCP. The performance results for a cabled network presented in [9] are comparable to ones obtained by a simulation model. However, since physical and medium access characteristics are not considered by TCP, transmission failures can be mistaken as network congestion in wireless networks. A false detection of congestion causes TCP to reduce its flow transmission rate and thus the overall wireless network performance.

This paper analyzes the application of a model based on max-plus algebra to study the behavior of TCP flow segments transmitted over a wireless network. This model is linear in max-plus algebra and complex topologies can be represented by a serial composition of simpler subsystems through a simple matrix multiplication as described in the paper. Some aspects of TCP behavior in wireless networks are also discussed as a background for a correct interpretation of results. Performance measures for the max-plus model are compared to ones obtained by discrete event simulation. Finally, the paper discusses some aspects of the parameter tuning and limitations of the max-plus model.

The paper is organized as follows. Section II introduces DES concepts, max-plus algebra and synchronization. Section

Manuscrito recebido em 15 de maio de 2005; revisado em 18 de novembro de 2005.

R. Andreola (rodrigo.andreola@vono.net.br) is currently with GVT. Rua Lourenço Pinto, 299 - Curitiba - PR - Brasil - 80010-160. R. Lüders (luders@utfpr.edu.br) and K. V. Fonseca (keiko@utfpr.edu.br) are both with Federal University of Technology-Paraná - UTFPR. Av. Sete de Setembro, 3165 - Curitiba - PR - Brasil - 80230-901.

III presents the max-plus model for TCP proposed by [9] for cabled networks and Section IV reviews some aspects of TCP behavior in wireless networks. Section V states the necessary parameters to evaluate the max-plus model for a wireless network. Section VI compares the results obtained by using the max-plus model to ones obtained by a simulation model in a wireless network and Section VII concludes the paper and presents future directions.

II. Max-Plus Algebra and Synchronization

A Discrete Event System (DES) is an event-driven system or a system whose state changes are characterized by the occurrence of events [10]. Such systems occur in many applications areas such as manufacturing, computer networks, logistic, among others. During the last decades many different approaches have been developed for modeling, analysis and control of DES [10]. These approaches are quite different according to the system aspect that should be captured. In particular, max-plus algebra is applied for modeling DES subject to synchronization phenomena [1]. These systems can be characterized by two basic operations: maximization and addition. Maximization is necessary for modeling an event that only occurs when a set of previous events have already occurred. For instance, the departure time of a trip that can only begins when all participants arrive. On the other hand, addition is necessary to compute time delays. For instance, a trip can take hours or days to arrive at its destination. Thus, the arrival time is obtained by adding a travelling time to the departure time. This algebraic structure is suited to model synchronization phenomena where events (departure and arrival of packets) should be delayed or synchronized according to the system modeling by addition and maximization operations, respectively. The max-plus algebra can thus capture temporal properties of systems only subject to synchronization. This basic idea can be generalized for other operations depending on which phenomenon have to be captured. For example, minimization can be used instead of maximization if a lower bound is required. Although operations can have different meanings for particular applications, a general algebraic structure can be defined. It is normally referred to as an idempotent semiring or dioid algebra.

Formally, an idempotent semiring or dioid denoted by $(D,\oplus,\otimes)$ is characterized by a set D with two operations (addition and multiplication) such that addition is associative, commutative and idempotent, multiplication is associative (not necessarily commutative) and it distributes by left and right over addition. Moreover, there exist neutral elements in D for both operations. They are denoted by ε (null element) and e (unit element) such that $a\oplus\varepsilon=\varepsilon\oplus a=a$, and $a\otimes e=e\otimes a=a$. The null element ε is also absorbing for multiplication ($a\otimes\varepsilon=\varepsilon\otimes a=\varepsilon$).

The max-plus algebra $R_{\max}=(R\cup-\infty,\oplus,\otimes)$ is an idempotent semiring satisfying the above properties. It is defined in the set of real numbers R added by $\varepsilon=-\infty$ (minus infinity) with addition and multiplication being computed by maximization and usual sum, respectively [1]. Naturally, the basic definition can also be extended to the set of matrices with components in $R_{\max}$. In this case, addition and multiplication of matrices are similarly defined as in the classical matrix theory but replacing the usual sum and product of elements by maximization and usual sum, respectively. In addition, the null matrix $\boldsymbol{\varepsilon}$ is defined by all components equal to ε and the identity matrix I has diagonal elements equal to e and ε otherwise.

It can be shown that many systems can be modeled by representing time dates of event occurrences as a linear function (in max-plus algebra) of previous event occurrences. This yields to a max-plus linear system [1]. Many works can be found in the literature regarding modeling and control of max-plus linear systems [1] and [11]. In the following, a max-plus linear model is used to compute departure and arrival times of packets in TCP flow segments transmitted across routers in a network.

III. A Max-Plus Model for TCP

The basic idea to characterize departure and arrival times of packets in a computer network through a max-plus model is based on a premise that packets should wait in a queue until a router is free to send them. A packet in a queue can only be sent after all previous packets have been sent. This is characterized by maximization between the arrival time of a packet in a queue and the departure time of the last packet arrived before it. Moreover, a transmission time between routers should also be considered. It is added to the departure time of a packet in a router to compute its arrival time in the next router. Finally, TCP controls how packets are sent to the network from a transmission source. The parameters of this model are processing times for each router (aggregate service time), transmission times between routers (latency time) and a current TCP window size that controls the number of packets sent to the network.

A. Network representation

This section presents the max-plus model proposed by [9] where further details can be found. In this model, a network is represented by K routers connected one after other as depicted in Fig. 1. The first router (source) sends a reference data flow to be transferred to the last router (destination) using TCP as transmission control protocol. Although TCP allows bidirectional connections between nodes, the max-plus model assumes a unidirectional connection by considering that a similar reverse path exists.

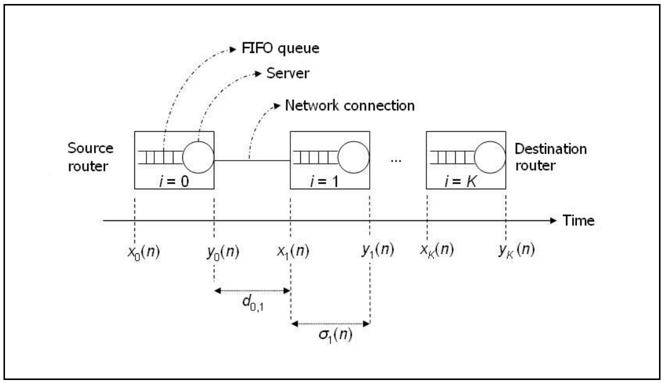

Fig. 1. A network as a tandem connection of K routers.

Routers in Fig. 1 are identified by $i = 0,1,2,...,K$ with $i = 0$ corresponding to the source router and $i = K$ to the destination router. Each router is modeled by a FIFO queue served by only one server which can attend not only packets from the reference flow but from cross traffic. The service time for processing a packet includes thus a processing time for cross traffic yielding to an aggregated service time.

A packet n arriving in a router i spends an aggregated service time $\sigma_i(n)$ computed since its arrival until its departure from a router. It includes both packet processing time and time necessary to process crossing traffic. Moreover, routers are connected each other by circuits with a defined latency time $d_{i,j}$ from router i to j. The number of packets sent by the source router to the network is controlled by adjusting a TCP window size [12].

This model also considers that an acknowledge packet *ACK* for delivery confirmation follows a different path from destination to source as if it are directly connected through a private circuit. Thus, *ACK* packets do not follow a reverse path regarding to the original TCP connection but a different path between destination and source. Although this assumption simplifies what really occurs in practice, it allows represent time transmission of *ACK* packets as a simple time latency $d_{K,0}$.

B. Equations for departure and arrival times of packets

The model used to quantify temporal performance of a TCP flow is composed by a set of equations that determines departure times for each packet of the reference flow in each router. These departure times of packets are computed by considering a dynamical behavior of TCP window, a service time for each router, and the transmission time latency between routers.

A packet n arrives in a router i at time $x_i(n)$ and leaves it at time $y_i(n)$. The aggregate service time $\sigma_i(n)$ corresponds to the difference between arrival and departure times of a packet n in router i. Therefore, $y_i(n) = x_i(n) + \sigma_i(n)$.

Since TCP dynamically controls the amount of packets sent to the network by adjusting its transmission window size, a new window size is computed when an *ACK* is received (a delivery confirmation) or when a timeout occurs. A TCP window size v_n is computed according to the information regarding packet n (delivery success or timeout). This information is then used to transmit a packet n+1. Indeed, v_{n-1} is the window size used to transmit a packet n.

By considering a previously known sequence of values $\{v_n\}$ and an aggregate service time $\sigma_0(n)$ equal to zero (by default), output times for a packet n in all routers can be stated in max-plus algebra by (1) and (2).

$$y_0(n) = y_K(n - v_{n-1}) \otimes d_{K,0} \tag{1}$$

$$y_i(n) = [y_{i-1}(n) \otimes d_{i-1,i} \oplus y_i(n-1)] \otimes \sigma_i(n) \quad 1 \le i \le K \tag{2}$$

The output time of packet n in router i is given by $y_i(n)$; v_{n-1} is the actual TCP window size for packet n; $d_{K,0}$ is the transmission time between destination and source routers (in particular, $d_{i-1,i}$ is the transmission time between router i and i-1); $\sigma_i(n)$ is the aggregate service time for packet n in router i.

Therefore, the time instant when a packet n is sent to the network is determined by (1), and a new packet can only be sent to the network after receiving an *ACK* from a previous sent packet. The actual TCP time window v_{n-1} determines which previous sent packet should be used. For instance, if n=5 and v_{n-1}=2, then packet number 5 should be sent to the network when an *ACK* for packet number 3 is received. Since $\sigma_0(n) \equiv 0$ by default, a new packet n is sent to the network as soon as an *ACK* arrives. It corresponds to the output time of packet number 3 from the last router added by *ACK* latency.

The network behavior is modeled by (2) since it determines output times of packets leaving each router. The network behavior is represented by transmission times $d_{i-1,i}$ between routers as well as by the aggregate service time $\sigma_i(n)$ of each router. The output time of a packet n leaving a router i in (2) is obtained by adding (through max-plus multiplication) its respective aggregate service time $\sigma_i(n)$ to the maximum time (obtained by max-plus addition) between its arrival time $y_{i-1}(n) \otimes d_{i-1,i}$ in router i and the departure time $y_{i-1}(n-1)$ of a previous packet n-1 in router i. The arrival time $y_{i-1}(n) \otimes d_{i-1,i}$ of n in router i is obtained by summing (max-plus multiplication) its output time from a previous router i-1 to the latency time $d_{i-1,i}$ between routers i-1 and i.

Transmission times between routers are typically constants in a network, and the aggregate service time can be determined according to router features and current cross traffic. Therefore, output times of packets in the reference flow can be obtained by a recursive form (2) for each router in the network.

The output times $y_i(n)$ of (2) can be grouped into a row vector $Y(n)$ given by (3) whose components represent the output times of a packet n in each router $i = 1,2,...,K$.

$$Y(n) = (y_1(n), y_2(n), \dots, y_K(n)) \tag{3}$$

A similar procedure can be applied to obtain a column vector $Z(n)$ defined as a composition of vectors $Y(n)$ for different values of *n* as shown in (4).

$$Z(n) = (Y(n) \,|\, Y(n-1) \,|\, ... \,|\, Y(n-w^*-1))^{\mathrm{T}}, \quad Z(n) \in \mathrm{R}_{\max}^{m} \quad (4)$$

The value w^* is the largest observed TCP window size and $m=Kw^*$. The vector $Z(n)$ contains temporal information about current and previous packets limited to the largest TCP window size for all routers. It can be obtained from $Z(n-1)$ by a suitable matrix multiplication (in max-plus algebra) as given by

$$Z(n) = A_{v_{n-1}}(n) \otimes Z(n-1) \quad n \geq 1 \quad (5)$$

with $Z(0) = (0,0\ldots,0)^{\mathrm{T}}$.

The matrix $A_{v_{n-1}}(n) \in \mathrm{R}_{\max}^{m \times m}$ depends on the model parameters (aggregate service and latency times) as well as the current TCP window size v_{n-1}. It can be obtained by expanding each component of vectors $Z(n)$ and $Z(n-1)$ according to (1) and (2).

By inspecting $A_{v_{n-1}}$ in more detail, we obtain a suitable partition into matrix blocks of dimension $K \times K$ [9]. For a current TCP window size $v_{n-1}=1$, the partitioned form is given by (6).

$$A_1 = \begin{pmatrix} M_1 \oplus M_2 & \boldsymbol{\varepsilon} & \cdots & \boldsymbol{\varepsilon} \\ I & \boldsymbol{\varepsilon} & \cdots & \boldsymbol{\varepsilon} \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{\varepsilon} & \boldsymbol{\varepsilon} & I & \boldsymbol{\varepsilon} \end{pmatrix} \quad (6)$$

Matrices M_1 and M_2 are given by (7) and (8) below, I is a K dimension identity matrix, and $\boldsymbol{\varepsilon}$ is a K dimension null matrix.

$$[M_1(n)]_{ij} = \begin{cases} \sum_{k=j}^{i} \sigma_k(n) + \sum_{k=j}^{i-1} d_{k,k+1} & i \geq j \\ \varepsilon & i < j \end{cases} \quad (7)$$

$$[M_2(n)]_{ij} = \begin{cases} \sum_{k=1}^{i} (d_{k-1,k} + \sigma_k(n)) + d_{K,0} & j = K \\ \varepsilon & j < K \end{cases} \quad (8)$$

Moreover, the partition changes according to the current TCP window size. However, only the position of M_2 changes according to the examples below for $v_{n-1}=2$ and $v_{n-1}=w^*$, respectively.

$$A_2 = \begin{pmatrix} M_1 & M_2 & \cdots & \boldsymbol{\varepsilon} \\ I & \boldsymbol{\varepsilon} & \cdots & \boldsymbol{\varepsilon} \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{\varepsilon} & \boldsymbol{\varepsilon} & I & \boldsymbol{\varepsilon} \end{pmatrix} \qquad A_{w^*} = \begin{pmatrix} M_1 & \boldsymbol{\varepsilon} & \cdots & M_2 \\ I & \boldsymbol{\varepsilon} & \cdots & \boldsymbol{\varepsilon} \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{\varepsilon} & \boldsymbol{\varepsilon} & I & \boldsymbol{\varepsilon} \end{pmatrix} \quad (9)$$

Therefore, a network can be characterized by output times of packets in each router through a linear recursive equation (in max-plus algebra) given by (5). In addition, the matrix $A_{v_{n-1}}(n)$ contains information about various aggregate service and transmission times between routers according to (7) and (8). It is basically composed by the network information that can even be determined before a TCP transmission begins. This linear model allows obtain more complex topologies by composing simple subsystems in series by multiplication (in max-plus algebra) of subsystem matrices.

In Section V, we propose to evaluate the max-plus model for a wireless network. But in the next section, we previously review some aspects of TCP behavior in wireless networks.

IV. Wireless Network Performance

Several works reported that wireless networks (particularly *ad hoc* ones) present poor performance when compared to wired networks [13] and [14]. Multiple accesses to a shared transmission medium and the number of control packets needed for collision control (as specified by the standard IEEE 802.11 [15], for example) are among the factors affecting the performance of a wireless network. In addition, TCP protocol also degrades the network performance since it identifies transmission failures as network congestion, as discussed in the next subsection.

A. TCP limitations for wireless networks

In traditional wireless (infrastructure mode) networks, changes on network topology or transmission failures are not usual. The TCP protocol proved to be robust at this particular scenario since it can adapt its transmission rate under network congestion events. Nevertheless, in *ad hoc* networks, topology changes and transmission failures are very usual and cannot be correctly detected by mechanisms of TCP [16] – they cannot distinguish a packet transmission failure from a network congestion event. Therefore, TCP performance tends to degrade where transmission failures regularly occur, that is, typical *ad hoc* wireless networks with mobile users. The performance of a TCP flow depends on its transmission window size and retransmission timers. Based on the reception or not of acknowledgment messages of previous flow segments already sent, TCP mechanisms adapt the flow transmission rate. This rate control aims to decrease congestion events and provide fairness on the network resource sharing. The rate control sets transmission window sizes and time between retransmission messages.

In wireless networks, transmission interruptions are usually caused by changes on the transmission channels. These changes and related interruptions typically occur at short time intervals within data packets or acknowledgment messages can be lost. Assuming changes occurring at short time intervals, fast retransmission of lost packets increases the probability of the retransmission correctly reach its destination. However, such behavior is not considered by most of usual TCP mechanisms. Based on this fact, [14] present a model to compute the expected data transfer rate for *ad hoc* wireless networks. In this model, the transfer rate is the maximum one obtained in a TCP transmission as a function of the number of link hops between source and destination.

Evaluation of TCP performance under several connection scenarios have motivated researchers to develop new TCP mechanisms for overcoming performance degradation in *ad hoc* wireless networks [17]. Examples of these mechanisms designed to wired networks are: delay on decreasing a transmission window due to loss of a single TCP segment [12] or a quick return to a previous window size after a burst of segment losses caused by a congestion event [18].

The effects mentioned above for wireless networks will be present in our simulation models. However, they should be also captured by the max-plus model if performance results have to be compared as in the following.

V. The Max-Plus and Simulation Models for a Wireless Network

The max-plus model can provide time instants for all packets in a network as depicted in Fig. 1 by specifying proper parameters. In the literature, performance measures were obtained by using the max-plus model and compared to ones obtained with a simulation model configured to represent a cabled network [9]. In the following, we propose adopting the same max-plus model with modified parameters, but simulation is configured for a wireless network. In this case, many aspects discussed in the previous section should be dealt.

We consider a wireless *ad-hoc* network with a topology similar to the one showed in Fig.1. This network uses a multi-hop mechanism for communicating two nodes which means that intermediary nodes can be used for routing of packets. The simulation model was implemented using a default routing protocol (Destination-Sequenced Distance Vector - DSDV) for static nodes in *ad-hoc* networks. Although node mobility could be taken in simulation, it was not considered here.

Simulation parameters are shown in Table I. They are typical ones used by ns-2 for network IEEE 802.11 and TCP.

The results were obtained by using a toolbox for max-plus computations [19] and ns-2 simulator [20]. Further details can be found in [21].

TABLE I
Simulation parameters.

Parameter	Value
Link protocol for wireless network	802.11
Transmission rate wireless network	2 Mbps
Router protocol	Destination-Sequenced Distance Vector - DSDV
Node distance	250 m
TCP data packet size	1210
TCP ACK packet size	40
Transmission queue size	50 packets

VI. Results

The results were obtained for two cases considering a wireless *ad hoc* network composed by five nodes.

A. Case I

The parameters used for max-plus model are shown in Table II. They are specified according to the simulation parameters.

TABLE II
Parameters for the max-plus model (Case I).

Parameter	Value
Destination router index (K)	4
Latency between routers ($d_{0,1}$, $d_{1,2}$, $d_{2,3}$, $d_{3,K}$)	10ms
Aggregate service time for routers ($\sigma_i(n)$)	5ms
ACK return time ($d_{K,0}$)	1ms

Routers are numbered from 0 to K=4. The latency of 10ms between routers was estimated by simulating a typical TCP transmission. The aggregate service time of 5ms for routers was calculated by considering links of 2Mbps and packets of size 1250 bytes (10000 bits/2Mbps=5ms to be processed). A value of 1ms for *ACK* return was considered for both models (max-plus and simulation).

The max-plus model also requires a TCP window dynamics. In this work, the TCP window starts with value 1 and changes according to the Tahoe method [12] without slow-start. However, the window is limited to a constant value as soon as traffic congestion is detected. This is accomplished by using a Rate Based Loss Detection method according to [9].

A total simulation time of 200s was considered. The first half of simulation time was used for routing protocol stabilization. Therefore, a TCP transmission is only started at time 100s avoiding routing traffic interference.

Results for the max-plus model and simulation can be compared by means of Table III whose transmission rates are given in packets per second. They are values of mean transmission rate obtained during TCP transmission until reception of the *ACK* for the packet in the first column of

Table III.

It can be seen from Table III that max-plus values are quite different from those obtained by simulation. While the aggregate service times for routers in max-plus model are set to represent a transmission rate of 2Mbps, the effective transmission rate observed by simulation between two nodes is quite smaller.

TABLE III
COMPARISON OF TRANSMISSION RATES (CASE I).

Packet	max-plus (pct/s)	simulation (pct/s)
5	26,60	20,08
10	38,61	20,58
50	79,37	20,89
100	111,35	20,17
250	151,69	19,66
500	172,53	19,85
1000	185,25	19,58

This can be explained by a shared transmission rate among nodes due to concurrency of transmission channel. Since the max-plus model does not consider a transmission concurrency it is not able to capture this behavior without a suitable tuning of its parameters. Moreover, it does not consider a proper dynamics of TCP window size. The protocol IEEE 802.11 drops a packet when many channel reservations fail or when no *ACK* is received after many transmissions. These dropped packets are interpreted as congestion by TCP protocol which decreases its corresponding window size. However, TCP window size remains unchanged in the max-plus model after reaching its largest size without congestion.

B. Case II

The parameters of max-plus model are changed for capturing a more realistic behavior. The max-plus model has two potential parameters to be changed: time latency and aggregate service time. Since time latency only depends on the channel speed, the aggregate service time can be increased to represent channel concurrency. A new value of 15ms for the aggregate service time was been experimentally obtained as shown in Table IV.

TABLE IV
PARAMETERS FOR THE MAX-PLUS MODEL (CASE II).

Parameter	Value
Destination router index (K)	4
Latency between routers ($d_{0,1}$, $d_{1,2}$, $d_{2,3}$, $d_{3,K}$)	10ms
Aggregate service time for routers ($\sigma_i(n)$)	**15ms**
ACK return time ($d_{K,0}$)	1ms

Although this value is experimental and obtained for matching simulation results, it shows that a max-plus model is able to represent a wireless network behavior. Moreover, dynamics for TCP window size are also been modified. Instead of using a bounded time window size limited by traffic congestion, a window size of 2 (constant) packets was used.

Although it seems to be quite restrictive, a window size of 2 packets is suitable for a wireless network of four hops since no more than two nodes can transmit at the same time. For instance, if node 0 is transmitting, then node 1 cannot transmit since it is receiving. Node 2 cannot transmit either since it may cause collision in the reception of node 1. Therefore, only node 3 and 1 can transmit at the same time.

The results are now drastically changed by modifying the aggregate service time as shown in Table V.

TABLE V
COMPARISON OF TRANSMISSION RATES (CASE II).

Packet	max-plus (pct/s)	Simulation (pct/s)
5	15,77	20,08
10	16,53	20,58
50	19,05	20,89
100	19,42	20,17
250	19,65	19,66
500	19,72	19,85
1000	19,76	19,58

Note that transmission rates stabilize on 19 pct/s after packet number 50. This shows that changes made on parameters of max-plus model are able to reflect the network congestion by limiting its effective transmission rate.

Values for transmission rates show little difference (about 1%) by comparing max-plus and simulation models for this TCP transmission. It thus suggests that a max-plus model can be used for modeling a TCP transmission over a wireless network since suitable parameters are used. Basically, an effective transmission rate should be correctly set as well as a proper change in TCP window size.

VII. CONCLUSION

This paper presented a detailed analysis of using a max-plus model for TCP found in the literature for a wireless network. Although the max-plus model was proposed in the literature to be used for a cabled network, it can also be used for a wireless network. The point is that aggregate service time plays an important role to represent a decreasing transmission rate due to the transmission concurrency between nodes. The aggregate service times of routers can be increased to proper reflect this effect while keeping latency time unchanged. In fact, latency times between nodes only reflect characteristics of the channel used for transmission. Various effects observed in wireless networks can thus be properly modeled. Note that the max-plus model requires a reduced number of parameters to be configured and it is linear in max-plus algebra. This could be an effective when dealing with large networks. Although the actual aggregate service time was experimentally obtained, it can be set in the future from the operational network conditions.

ACKNOWLEDGEMENT

This work was supported by *Fundação Araucária* and *Fundo Setorial de Infra-Estrutura* (CTINFRA)-MCT/CNPq under grant 233/07-11015 (FUNTEF-PR).

REFERENCES

[1] F. Baccelli, G. Cohen, G. Olsder, and J. Quadrat, *Synchronization and Linearity: An Algebra for Discrete Event Systems*. New York: John Wiley, 1992.
[2] J. Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Lec. Notes in Comp. Science Series, n. 2050, New York: Springer, 2001.
[3] M. Daniel-Cavalcanti, “Modelagem e Determinação de Parâmetros de Desempenho de Redes de Comunicações Através da Álgebra de Dióides”, PhD dissertation, FEEC, Unicamp, Campinas, SP, 2008 (in Portuguese).
[4] *IETF RFC: 793- Transmission Control Protocol*, Darpa, 1981.
[5] T. Lakshman and U. Madhow, “The performance of TCP/IP for networks with high bandwidth-delay products and random loss”, *IEEE/ACM Transactions on Networking*, vol. 5, no. 3, pp. 336-350, 1997.
[6] M. Mathis, J. Semke, J. Mahadavi, and T. Ott, “The macroscopic behavior of the TCP congestion avoidance algorithm”, *Computer Communications Review*, vol.27, no.3, pp. 67-82, 1997.
[7] J. Padhye, V. Firoiu, and D. Towsley, “A stochastic model of TCP Reno congestion avoidance and control”, Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.
[8] K. Hwangnam and J. C. Hou “Network calculus based simulation for TCP congestion control: theorems, implementation and evaluation”, in *Proc. of INFOCOM 2004*, Hong Kong, 2004, pp. 7-11.
[9] F. Baccelli and D. Hong, “TCP is max-plus linear and what it tells us on its throughput”, in *Proc. of ACM SIGCOMM'00*, Stockholm, 2000, pp. 219-230.
[10] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. New York: Springer, 2nd ed., 2007.
[11] C. A. Maia, R. Lüders, R. Santos-Mendes, and L. Hardouin, “Estratégias de controle por modelo de referência de sistemas a eventos discretos max-plus lineares”, *Revista Controle & Automação*, vol.16, no.3, pp. 263-278, 2005.
[12] V. Jacobson, “Congestion avoidance and control”, in *Proc. of ACM SIGCOMM'88*, Stanford, 1988, pp. 314-329.
[13] J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris, “Capacity of ad hoc wireless networks”, in *Proc. of the 7th Int. Conference on Mobile Computing and Networking*, New York, 2001, pp. 61-69.
[14] G. Holland and N. Vaidya, “Analysis of TCP performance over mobile ad hoc networks”, in *Proc. of the 5th Int. Conf. on Mobile Computing and Networking*, New York, 1999, p. 219-230.
[15] IEEE. (2009, Sep. 3). IEEE Std 802.11: Wireless Local Area Networks [Online]. Available: http://grouper.ieee.org/groups/802/11/.
[16] IETF. (2009, Sep. 3). RFC 2001: TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms [Online]. Available: http://rfc.dotsrc.org/rfc/rfc2001.html.
[17] A. A. Hanbali, *et al*, “A Survey of TCP over ad hoc networks”, *IEEE Communications Surveys & Tutorials*, vol. 7, no. 3, pp. 22-36, 2005.
[18] S. Floyd, *et al*. (2009, Sep. 2). RFC 3782 - The NewReno Modification to TCP's Fast Recovery Algorithm [Online]. Available: http://www.faqs.org/rfcs/rfc3782.html.
[19] J. Stanczyk, *Max-Plus Algebra Toolbox for Matlab and GNU Octave v.1.5*, Otto-von-Guericke Universität, Magdeburg, 2005.
[20] K. Fall and K. Varadhan. (2009, Aug. 26). The ns Manual [Online]. Available: http://www.isi.edu/nsnam/ns/ns-documentation.html.
[21] R. Andreola, “Análise de um Modelo TCP Max-plus Aplicado a uma Rede Sem Fio”, Master dissertation, CPGEI, UTFPR, Curitiba, PR, 2007 (in Portuguese).

Rodrigo Andreola was born in Curitiba, PR, on July, 1977. He received his Bsc in Computer Science from UFPR in 2001 and Msc in Electrical Engineering from UTFPR in 2007.
He is with GVT since 2000, working with information technology and product development for telecom industry.

Ricardo Lüders was born in Campinas, SP, on April, 1966. He received his Bsc, Msc and PhD in Electrical Engineering from the Department of Electrical and Computer Engineering of UNICAMP in 1989, 1992 and 2001, respectively.
He is professor in the Federal University of Technology-Paraná since 1997. Actually, he is with the University of Michigan – EUA for a post-doc year. His teaching and research activities regards Electrical and Computer Engineering area, and particularly Discrete Event Systems, including operational research, max-plus algebra, Petri nets and hybrid systems.

Keiko V. Fonseca was born in São Paulo, SP, on January, 1960. She received her Bsc, Msc and PhD in Electrical Engineering from the UFPR, UNICAMP and UFSC in 1985, 1998 e 1997, respectively.
She is professor in the Federal University of Technology-Paraná (UTFPR) since 1988 and is currently dean of the Computer Engineering College and member of the Electronics Engineering Advisory Council of the UTFPR. Her research activities are in Electrical and Computer Engineering, particularly in Real Time Communication Systems, Digital Television, and Computer Networks. She is member of the IEEE Communications Society, IEICE (Japan) and SBC (Brazilian Computer Society).