Performance Bounds in Coupled Processor Systems

Christian Vitale
Institute IMDEA Networks
christian.vitale@imdea.org

Gianluca Rizzo
HES SO Valais, Switzerland
gianluca.rizzo@hevs.ch

Balaji Rengarajan
Accelera MB, CA, USA
balaji.rengarajan@acceleramb.com

ABSTRACT

We consider queuing systems with coupled processors, where the service rate at each queue depends on the set of active queues in the system. In general, the queue lengths of such systems, that we call Coupled Processor Systems (CPSs), are modeled through complex Markov Chains whose steady state distributions are known only for two-queue systems. For larger systems, the performance modeling has either been based on very large sets of simulations, or on neglecting altogether the effect of the couplings on system dynamics.We propose a new approach, based on a worst case analysis of system dynamics, and which does not require simulations for its parametrization. We derive new sufficient conditions for stability of a CPS, and bounds on backlog and packet delay. We assess numerically our results, showing that considering those dynamics of the system induced by coupling substantially improves resource allocation in a CPS.

1. INTRODUCTION

We consider a system of parallel queues, for which the service rate that the incoming traffic receives depends on the set of nonempty queues in the system. Such model, known in the literature as "Coupled Processors System" (CPS)[14], arises in several contexts, in which coupling typically derives from the sharing of common resources. In wireless communications, CPS hes been proposed to model the effects on performance of the complex interdependence among nodes due to the shared medium, as well as of interference [3]. Indeed the increase in capacity demand in wireless access is bringing up base stations and access points densities, increasing the impact of interference on performance. In LTE, the use of a low reuse factor has been proposed in order to increase the capacity to the user [13]. This makes it essential to take into account in network planning the impact of the resulting interdependency among base stations service times. Other applications of the coupled processors model are in the analysis of performance of clusters of servers, of computing services based on a virtualization of the underlying infrastructure, of network of processors, and in any system of servers in which the sharing of a common resource induces a correlation between the performance of the servers [4, 5, 3]. In such systems overprovisioning is often the main approach adopted to minimize the effect of coupling.

In the present work, we focus on CPSs where the service rate of each queue is a decreasing function of the number of active queues in the system. This class of CPSs applies to many problems of practical importance (including all of the above mentioned examples) and it has been widely studied in the past, in the context of wireless networks and of bandwidth sharing in packet networks. A trivial way of analyzing such systems is to assume a static, worst-case scenario, where service rates are those of a system in saturation, with all queues active. This leads to heavily pessimistic results, often of little practical interest. Research efforts have focused therefore on the (very difficult) problem of capturing the effects of system dynamics on performance. [8, 11] derived closed-form necessary and sufficient conditions for the stability of a CPS composed by two nodes with one class of traffic each. Both works assume Poisson arrivals and exponential service times. [4] derives a similar result, assuming heavy tailed distribution for file sizes. [5] provides a method to know if a particular configuration of the CPS is stable, based on the steady state probabilities of related Markov chains, and under Poissonian traffic assumption. The drawback of this method is that it is based on a number of simulations for the derivation of the steady state probabilities which grows factorially with the size of the problem.[12] applies this method to a scenario with three nodes and a single class of traffic for each of them. The approach in [14] is based on the stochastic characterization of bounds on the moment generating function for the queue lengths at the nodes of the CPS. [2] proposes approximation methods for the derivation of sufficient conditions for stability, which assume that all nodes but one are saturated.

As the complexity of all available results is a factorial or exponential function of the number of nodes, the derivation of nontrivial and computationally feasible performance bounds for CPS is still an open problem. Indeed, many of the available results involve computationally heavy simulations or they are based on conservative assumptions on the system. In the present paper, we propose a fully analytical approach for performance analysis of a generic N-nodes CPS, showing how to derive the main performance bounds of the system. For a given CPS, our method is based on the derivation of a set of networks whose performance bounds hold also for the CPS, and which can be analyzed using standard Network Calculus results. Summarizing, in this paper:

- We present an analytical method for computing sufficient stability conditions for a generic N nodes CPS when arrivals are leaky bucket constrained;
- We show how to derive hard bounds on backlog at each node and, when nodes are FIFO, on maximum packet delay;
- We assess numerically our results by applying them on a spe-

cific problem in wireless communications. Furthermore, we show that considering the dynamics of the incoming traffic at a CPS helps improving substantially the resource allocation of the system.

The paper is organized as follows. In Section 2 we present the system model used in this paper, and some important Network Calculus concepts. Section 3 presents our method for the derivation of performance bounds in coupled processors systems, and in Section 5 we show on a wireless scenario how to derive practical results, assessing their performance on some numerical examples. We conclude our paper and discuss future directions of research in Section 6.

2. MODEL AND ASSUMPTIONS

2.1 System Model

We consider a system of N parallel queues, where each queue receives traffic from one or more fresh sources (i.e. residing out of the system). We assume such queues are served by work conserving schedulers. We define as the *state* of the system at time t the array $\mathbf{I}(t) = (I_1(t), I_2(t), ..., I_N(t))$ where $\forall i$, $I_i(t)$ is 0 if the queue at the $i-$th node is empty at time t, and 1 otherwise. $\forall i$ we assume that the instantaneous service rate at queue i, $R_i(t)$, is determined only by the state of the system at time t, i.e. $R_i(t) = R_i(\mathbf{I}(t))$. We call such a system a *Coupled Processor System* (CPS).

In the present treatment we consider only monotonic CPSs. i.e such that if $\mathbf{I}_1$ and $\mathbf{I}_2$ are two different states of the system such that $\mathbf{I}_1 \leq \mathbf{I}_2$, then $\forall i,\ R_i(\mathbf{I}_1) \geq R_i(\mathbf{I}_2)$. In Section 4 we outline how to extend our results to non-monotonic CPSs.

Without loss of generality, we assume arrivals to be *packetized*, with a finite number of packet sizes. We consider that no losses are present at queues (buffers of infinite capacity).

We assume the traffic from fresh sources to be constrained by a *leaky bucket arrival curve* [6]. This implies that if $A(t)$ is the cumulative arrival function for a given traffic source for the time interval $[0, t]$, than for any $[t_1, t_2]$, $A(t_2) - A(t_1) \leq \rho(t_2 - t_1) + \sigma$. ρ, σ are the *leaky bucket rate* and the *burstiness*, respectively, for the considered source. Such assumption on traffic covers a large spectrum of practical settings, as it allows us not to make any assumption on traffic statistics. Indeed, in real settings any source is constrained by some form of leaky bucket arrival curve, (e.g. due to limitations of the application generating the traffic, to the bit rate of the connection, and so on), possibly by means of some conservative assumptions on the statistics of the traffic.

2.2 Network Calculus Basics

In this section we introduce some Network Calculus concepts and results we have used in this paper. Network Calculus is a min-plus system theory for deterministic performance analysis of a queuing system [6]. It provides tools for the derivation of bounds to backlog and packet delay in a network. We first present the *continuous data scaling block*, originally introduced in [10]. We use a slightly different version of it:

DEFINITION 2.1 (CONTINUOUS DATA SCALING BLOCK). *For any time t_1, $t_2 \geq 0$, with $t_2 \geq t_1$, assume $a \geq 0$ is the amount of bits arrived at a node in the time interval $[t_1, t_2]$. The node is a Continuous Data Scaling Block, with scaling value $S \in \mathbb{R}^+$ if the amount of bits at its output during the same time interval is $S \cdot a$.*

The data scaling block makes it possible to model transformation processes which alter not only the timings of data arrivals, but also the total amount of data arriving, like data processing, encoding/decoding, and so on. Another concept we use is the one of the *policer*:

DEFINITION 2.2 (POLICER). *A policer with policing function $\gamma(t) \in \mathcal{F}$ is a processing device such that, for any arbitrary input traffic, forces $\gamma(t)$ as arrival curve for the output traffic.*

Note that the definition does not specify what happens to the part of the input traffic exceeding $\gamma(t)$. In what follows we consider *unbuffered* policers, which discard non-conformant traffic.

Finally, we adopt the following definition of stability:

DEFINITION 2.3 (STABILITY). *Let us consider a system of N queues, and for every queue $i \in [1, ..., N]$ and any time $t \geq 0$ let $q_i(t)$ be the backlog at queue i at time t. The system is stable if $\forall i \in [1, ..., N]$ it exists a finite $\Gamma_i > 0$ such that for any arrival pattern from traffic sources, $sup_{t \geq 0} q_i(t) \leq \Gamma_i$*

In general, sufficient conditions for stability imply constraints on the fresh traffic arrivals (e.g. on leaky bucket parameters) and on the network (service rates). If the condition on the backlog is satisfied only for some queues, we say the system is *partially stable*.

3. A NETWORK CALCULUS APPROACH TO CPS ANALYSIS

A primary issue in coupled processors systems is to determine what are the sufficient conditions for the system to be stable, in order to be able to derive bounds on packet delay and on backlog. In this section we illustrate a method for the derivation of these bounds, based on a worst-case analysis. Our method is based on constructing, from a given CPS, a set of feed forward networks. Each of these networks is such that their partial stability over a specific subset of their nodes implies the stability of the original CPS (we say that such networks *upper bound* the CPS). As we show, the structure of these networks is such that it can be easily analyzed by means of standard Network Calculus techniques, for the derivation of the main performance bounds.

We begin with the following result, which defines a sufficient condition for a class of networks to upper bound a CPS:

THEOREM 3.1 (UPPER BOUNDING NETWORK). *Consider a network with $N' \geq N$ queues, such that there is a one-to-one mapping between the queues of the CPS and a subset of N queues of the network. The mapping is such that each queue j in the subset has the same arrivals at any time t as its corresponding queue n_j in the CPS. Let $R_{n_j}(t)$ and $R_j(t)$ be the service rates at time t, respectively, at queue n_j and queue j. If at any time $t \geq 0$, for each node j of the network, it holds $R_j(t) \leq R_{n_j}(t)$, then the network upper bounds the CPS.*

PROOF. We prove that at any time $t \geq 0$, the length $q_{n_j}(t)$ of each queue n_j of the CPS is always smaller than the queue length $q_j(t)$ of the corresponding node in the network. If this is the case, a bound on the backlog of j also holds for the corresponding node of the CPS, so that the partial stability of the network implies the stability of the CPS. We prove $q_{n_j}(t) \leq q_j(t)$ by contradiction. Assume that t^* is the smallest time for which $q_{n_j}(t^*) > q_j(t^*)$ holds. As arrivals are the same in both queues, this implies that at time t^* a bit b of traffic has left queue j in the network, while is still being served at the corresponding queue n_j of the CPS. Assume the last bit served before b at node j has been served at $t^* - \epsilon_j$ in the network, and the last bit has been served at node n_j at time $t^* - \epsilon_{n_j}$. From the definition of t^*, $t^* - \epsilon_j \geq t^* - \epsilon_{n_j}$. This it means it exist an instant $t' \in [t^* - \epsilon_{n_j}, t^*)$, where $R_{n_j}(t') \leq R_j(t')$, which contradicts the assumptions. □

We now describe the structure of a network satisfying the conditions in Theorem 3.1, for the given CPS. The network has a feed

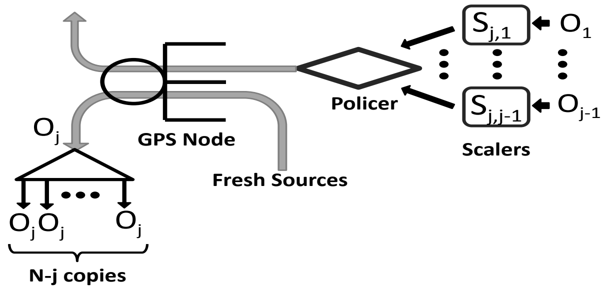


Figure 1: Structure of the j-th stage of the feed forward network.

forward topology with N stages. At each stage there is a work-conserving GPS node, with two queues and working at the fluid limit.
Let us label each queue of the CPS from 1 to N. We also label the GPS nodes of the network, in a way that $j \in 1, ..., N$ is the label of the j-th stage of the network, as well as of the GPS node in it. If $\mathbf{n} = (n_1, ..., n_N)$ is one of the $N!$ possible permutations of the labels of the CPS, **n** identifies a specific mapping which associates the j-th GPS node to the n_j-th node of the CPS.

Let us analyze the structure of the j-th stage of the feed forward network. As shown in Fig. 1, each traffic flow $O_k,\ k = 1, ..., j-1$ coming from one of the $j-1$ previous stages is fed to a dedicated scaling block, with scaling coefficent $S_{j,k}$.
The aggregate output of all the $j-1$ scaling blocks is fed to a policer, with policing function $\gamma(t) = (R^{up}_{n_j} - R^{sat}_{n_j})t$, where $R^{up}_{n_j}$ is the service rate of node n_j of the CPS when the active queues at the CPS are $n_j, n_{j+1}, ..., n_N$, and $R^{sat}_{n_j}$ when all queues of the CPS are active.
The output of the policer is fed to a queue of the GPS, while the other queue is dedicated to traffic from fresh sources. At any time t, we assume arrivals from fresh sources at stage j are the same as at the corresponding node n_j at the CPS. The total service rate of the GPS node is $R^{up}_{n_j}$, and the GPS weights are $w = R^{sat}_{n_j}/R^{up}_{n_j}$ for fresh traffic, and $1-w$ for traffic from the policer. As a consequence, it can be easily verified that traffic from the policer is never buffered at the GPS, so that stability of the network depends only on the queues dedicated to fresh traffic.
At the output of the GPS node, traffic coming from previous stages exits the network. The remaining traffic is fed to a node which produces $N-j$ exact replicas of the same traffic, introducing no delay. Finally, each replica is fed to one of the following stages.
We indicate with **S** the upper triangular matrix of the scaling coefficients for the whole network. For the given CPS, every couple (**n**,**S**) identifies a specific network with the structure we have illustrated. Similarly to what happens in a CPS, within each of these networks the queues for fresh traffic are coupled. Indeed, at any GPS node the instantaneous service rate for fresh traffic at a given time generally depends on a subset of the GPS nodes which are serving fresh traffic at that time. We indicate with $O_j(\mathbf{Y}(t))$ such a service rate at node j and time t, where the array $\mathbf{Y}$ is the *state vector* of the network, of length N, indicating the fresh traffic queues which are active at time t.
From the feed forward structure of the network, one can see that at any stage the service rate $O_j(\mathbf{Y}(t))$ at time t can be derived recursively from previous stages as:

$$O_j(\mathbf{Y}(t)) = R^{up}_{n_j} - F_j(\mathbf{Y}(t), \mathbf{S}), \tag{1}$$

where $F_j(\mathbf{Y}(t), \mathbf{S})$ is the output of the policer at stage j for the state vector $\mathbf{Y}(t)$:

$$F_j(\mathbf{Y}(t), \mathbf{S}) = min\left(\sum_{p=1}^{j-1} O_p(\mathbf{Y}(t))S_{j,p}, R^{up}_{n_j} - R^{sat}_{n_j}\right). \tag{2}$$

$F_j(\mathbf{Y}(t), \mathbf{S})$, which we call *rate impairment*, models the effect which fresh traffic at nodes at stages $1, ..., j-1$ have on the fresh traffic service rate at node j. Here we have used the fact that as fresh traffic is packetized, the instantaneous service rate for fresh traffic is also the instantaneous departure rate of this traffic from node j.
We now show how, by appropriately modulating rate impairments at each GPS node (via a proper choice of the scaling coefficients), it is possible to obtain a network which satisfies the hypothesis of Theorem 3.1, and which therefore upper bounds the CPS. The following theorem defines sufficient conditions on rate impairments for a network to upper bound the given CPS.

THEOREM 3.2. *It is given a CPS, and a (**n**,**S**) network structured as described, and let $\boldsymbol{Y}(t)$ be the state vector of the network at time t. Then the network upper bounds the CPS if, $\forall t \geq 0$, and for any node j the rate impairment $F_j(\boldsymbol{Y}(t), \boldsymbol{S})$ satisfies:*

$$F_j(\boldsymbol{Y}(t), \boldsymbol{S}) \geq R^{up}_{n_j} - R^{up}_{n_j}(\boldsymbol{Y}(t)) \tag{3}$$

where $R^{up}_{n_j}(\boldsymbol{Y}(t))$ is the service rate at node n_j of the CPS when the nodes of the CPS which are active are all those associated to active nodes in the network, plus nodes $n_{j+1}, ..., n_N$.

For the proof see Section 8.1. In the network we model these coupling effects with the presence of additional traffic into the queue affected, coming from other nodes in the network. More specifically, every time a node affects the service of another one in the given CPS, in the network there is some traffic going out from the node corresponding to the affecting node in the CPS, and entering the node which corresponds to the affected node in the CPS. As such network works at the *fluid limit* (i.e. traffic is served as if it was infinitely divisible), and as this additional traffic is present only when the affecting queue is active, we show that the performance of the affected queue at any time t depends on the occupancy state of the affecting queue at the same time instant, similarly to the original CPS.
One of the issues raised by Theorem 3.2 is whether it is always possible to build, for a given CPS, an upper bounding network. From the structure of the feed forward network, one can easily see that for every mapping **n** there are at least two choices for the scaling coefficients which brings them to always satisfy Theorem 3.2. The first choice consists in setting all scaling values to infinity, and it brings to heavily overestimate the effects of coupling on service rates. The second one is illustrated in Section 5.1, and, as we will show numerically, corresponds to a set of upepr bounding networks which more tightly follow rate variations of their associated CPS. As for each CPS there are $N!$ possible mappings, for each CPS there are at least $2N!$ upper bounding networks.

3.1 Sufficient conditions for stability of a CPS

In what follows, we consider the case in which, at each node of the CPS, the aggregate of the fresh traffic is constrained by a leaky bucket arrival curve [6]. It can easily be proven that similar results can be derived for other kind of arrival curves. The following theorem defines a set of sufficient conditions on the leaky bucket rates for the stability of the CPS, exploiting the properties of the upper bounding networks described in the previous section.

THEOREM 3.3. *Let us consider a CPS, where at each node $j = 1, ..., N$ fresh arrivals are constrained by leaky bucket arrival curves, with parameters (ρ_j, σ_j). If it exists at least one network $(\boldsymbol{n}, \boldsymbol{S})$ satisfying Theorem 3.2 for the given CPS, and such that at each stage $j = 1, ..., N$, ρ_j satisfies*

$$\rho_j \leq max\left(R^{sat}_{n_j}, R^{up}_{n_j} - \sum_{p=1}^{j-1} S_{j,p}\rho_{n_p}\right) \quad (4)$$

then the CPS is stable.

For the proof see Section 8.2. These sufficient conditions are computed from analyzing the upper bounding network by stages, and they derive from the imposing node stability at each GPS. As we can see from the expressions of these conditions, a trivial sufficient condition for stability is $\rho_j \leq R^{sat}_{n_j}, \forall j$, which corresponds to assuming each node always serves fresh traffic at the worst possible service rate due to coupling. The additional terms in the condition represent the improvement over such trivial conditions obtained by modeling the traffic at those nodes which affect the service rate of the considered node.

3.2 Bounds on Backlog and Virtual Delay

Once that for a CPS, we have built an upper bounding network $(\mathbf{n}, \mathbf{S})$ which is stable according to Theorem 3.3, then it is possible, by applying standard Network Calculus results on that same network, to derive upper bounds to packet delay and to backlog at each node of the network. Then, as the network satisfies also the sufficient conditions in Theorem 3.1, it can be easily shown that bounds for delay and backlog for each GPS queue dedicated to fresh traffic in such a network hold also for the corresponding queues in the CPS.

THEOREM 3.4. *Let us consider a CPS and an $(\boldsymbol{n}, \boldsymbol{S})$ upper bounding network which verifies the hypothesis of Theorem 3.3. Then a bound for backlog at each node n_j of the CPS, $j = 1, ..., N$ is given by:*

$$\sigma^*_{n_j} = \begin{cases} \sigma_{n_j}, & \text{if } \rho_{n_j} \leq R^{sat}_{n_j}, \\ \sigma_{n_j} + \frac{(\rho_{n_j} - R^{sat}_{n_j})\cdot(\sum_{k=1}^{j-1} S_{j,k}\cdot\sigma^*_{n_k})}{R^{up}_{n_j} - R^{sat}_{n_j} - \sum_{k=1}^{j-1} S_{j,k}\cdot\rho_{n_k}} & \text{otherwise.} \end{cases} \quad (5)$$

Moreover, if the nodes of the CPS are FIFO, a bound to packet delay at the same node is:

$$d_{n_j} = \begin{cases} \frac{\sigma_{n_j}}{R^{sat}_{n_j}}, & \text{if } \rho_{n_j} \leq R^{sat}_{n_j} \text{ and } \sigma_{n_j} \leq b_{n_j}, \\ T_{n_j} + \frac{\sigma_{n_j}}{R^{up}_{n_j} - \sum_{k=1}^{j-1} S_{j,k}\cdot\rho_{n_k}}, & \text{if } \sigma_{n_j} > b_{n_j}, \\ \frac{b_{n_j}}{R^{sat}_{n_j}} - \frac{b_{n_j} - \sigma_{n_j}}{\rho_{n_j}}, & \text{if } \rho_{n_j} > R^{sat}_{n_j} \text{ and } \sigma_{n_j} \leq b_{n_j}. \end{cases} \quad (6)$$

with:

$$b_{n_j} = \begin{cases} \infty, & \text{if } R^{sat}_{n_j} \geq R^{up}_{n_j} - \sum_{k=1}^{j-1} S_{j,k}\rho_{n_k}, \\ \frac{R^{sat}_{n_j}\sum_{k=1}^{j-1} S_{j,k}\sigma^*_{n_k}}{R^{up}_{n_j} - R^{sat}_{n_j} - \sum_{k=1}^{j-1} S_{j,k}\rho_{n_k}} & \text{otherwise.} \end{cases}$$

$$T_{n_j} = \frac{\sum_{k=1}^{j-1} S_{j,k}\cdot\sigma^*_{n_k}}{R^{up}_{n_j} - \sum_{k=1}^{j-1} S_{j,k}\cdot\rho_{n_j}}.$$

For the proof see Section 8.3. When several upper bounding networks are available for which the CPS is stable according to Theorem 3.3, then our results enable the optimization of a given function of these bounds, over the set of stable upper bounding networks.

3.3 Comparison with stochastic approach in [5]

In this section we investigate the relationship between our method and the one described in [5]. More specifically, we compare our sufficient conditions for stability with the necessary and sufficient conditions in [5], when independent Poisson arrival process with $\{\lambda_1, \lambda_2\}$ as parameter and exponentially distributed length for the arrivals with parameter $\{\mu_1, \mu_2\}$ are assumed for the system. As the two methods rely on different assumptions for the input traffic, and different notions of stability, no rigorous comparison is possible. In this section we try to get an intuition of how they relate to each other, by using a stochastic network calculus formulation.
The scenario we consider is a two nodes CPS. For larger number of nodes, a comparison would not be feasible as the method in [5] would require a very high number of simulations. Recalling the notation already used, R_i, with $i \in [1, 2]$, is the service rate node i of the CPS reserves to its own queue if the other node of the CPS is not active while R^{sat}_i is the service rate node i reserves to its own queue when both the node of the CPS are active. Applying the results of [5], necessary and sufficient conditions for stability of this system are:

$$\begin{cases} \frac{\lambda_1}{\mu_1} \leq R^{sat}_1 \\ \frac{\lambda_2}{\mu_2} \leq R_2 - \frac{\lambda_1}{\mu_1}\cdot\frac{R_2 - R^{sat}_2}{R^{sat}_1} \end{cases}$$

$$\begin{cases} \frac{\lambda_2}{\mu_2} \leq R^{sat}_2 \\ \frac{\lambda_1}{\mu_1} \leq R_1 - \frac{\lambda_2}{\mu_2}\cdot\frac{R_2 - R^{sat}_1}{R^{sat}_2} \end{cases} \quad (7)$$

In order to have an idea of how this result compares with ours, we use the Stochastic Network Calculus characterization of the Poisson inputs from [7]. If $A(t)$ is the cumulative arrival function, then a stochastic arrival curve can be defined for the inputs, with long term rate $\rho'_i = \frac{\lambda_i}{\mu_i - \delta}$, violation probability p, and burstiness $\sigma'_i = \frac{ln(p)}{-\delta}$, such that for any time interval $[s, t]$, with $s, t \geq 0$, it holds:

$$Pr\left(A(t) - A(s) > \frac{\lambda}{\mu - \delta}\cdot(t - s) + \frac{ln(p)}{-\delta}\right) \leq p.$$

with $\delta < \mu_i$. Now let us consider the same two nodes CPS, whose input are constrained by deterministic leaky bucket, with parameters (σ'_i, ρ'_i), and let us consider the two upper bounding networks we can build with our method. By applying Theorem 3.3, our sufficient conditions for stability are:

$$\begin{cases} \frac{\lambda_1}{\mu_1 - \delta} \leq R^{sat}_1 \\ \frac{\lambda_2}{\mu_2 - \delta} \leq R_2 - \frac{\lambda_1}{\mu_1 - \delta}\cdot\frac{R_2 - R^{sat}_2}{R^{sat}_1} \end{cases}$$

$$\begin{cases} \frac{\lambda_2}{\mu_2 - \delta} \leq R^{sat}_2 \\ \frac{\lambda_1}{\mu_1 - \delta} \leq R_1 - \frac{\lambda_2}{\mu_2 - \delta}\cdot\frac{R_2 - R^{sat}_1}{R^{sat}_2}. \end{cases} \quad (8)$$

By comparing (8) with (7) we can see that as the parameters δ and p of the stochastic arrival curve tend to zero, the system with deterministic leaky bucket inputs approximates better and better the one with Poisson arrivals, and our sufficient stability conditions tend towards those obtained with [5].

4. NON-MONOTONIC CPS

So far, we have considered the CPS to be monotonic. This assumption applies to a large set of practical problems modeled via coupled processors. In this section we outline how to extend our results to the general case in which the CPS is not necessarily monotonic.
Theorem 3.1 does not rely on the monotonicity of the CPS, and indeed the whole proof of this result holds also for nonmonotonic CPSs.
As for the upper bounding networks described in Section 3, for a generic CPS their structure remains the same as for the monotonic case. The only difference lies in the values taken, at stage $j = 1, ..., N$, by the total service rate of the j-th GPS nodes, by the GPS weights, and by the parameters of the policing function at that stage.
The definition of $R^{up}_{n_j}$, $R^{sat}_{n_j}$, as well as of $R^{up}_{n_j}(\mathbf{Y}(t))$ in $Theorem$ 3.2 assume that the system is monotonic. In the most general version, the definition of these parameters is the following. For a generic network $(\mathbf{n}, \mathbf{S})$, let $\mathcal{I}(n_j, n_{j+1}, ..., n_N)$ be the set of all possible states of the CPS in which nodes $n_1, ..., n_{j-1}$ are not active. Then:

- $R^{up}_{n_j} = \min_{I \in \mathcal{I}(n_j, n_{j+1}, ..., n_N)} R_{n_j}(I)$
- $R^{sat}_{n_j} = \min_{I \in \mathcal{I}(n_1, n_2, ..., n_N)} R_{n_j}(I)$
- Let us indicate with $\mathcal{U}(j, \mathbf{Y}(t))$ the subset of the nodes of the CPS composed by those nodes which are associated to active nodes in the network at time t, plus nodes $n_j, n_{j+1}, ..., n_N$. Then
$$R^{up}_{n_j}(\mathbf{Y}(t)) = \min_{I \in \mathcal{U}(j, \mathbf{Y}(t))} R_{n_j}(I)$$

Moreover, in the upper bounding networks described in Section 3, for every couple of GPS nodes (k, j), such that $2 \leq j \leq N$, $1 \leq k < j$, let us consider the corresponding nodes in the CPS. If whenever queue n_k gets active, the service rate at node n_j increases, then we put $S_{k,j} = 0$, or equivalently, we assume no traffic goes from stage k to stage j. In this way the upper bounding network we obtain is monotonic even if its associated CPS is not, and all the results on stability and bounds hold also for this case.

5. NUMERICAL RESULTS

Through the use of the method just introduced, each network (**n**,**S**) respecting respecting Theorem 3.2 leads to sufficient stability conditions for the CPS we are starting with. In other words, each network (**n**,**S**) respecting Theorem 3.2 identifies a set of values for the long term rates $(\rho_1, .., \rho_N)$ for which stability can be ensured. Furthermore, a bound on backlog and delay can be computed for each set of stable arrival rates $(\rho_1, .., \rho_N)$.

In this section we first present a set of scaling values S^w that respects Theorem 3.2. Then we show how to model a simple, but realistic, wireless scenario through the use of a CPS model. Then, we compare the analytical sufficient condition for stability and the bounds for packet delay with the set of achievable rates and per-packet delay obtained through the simulation of the wireless scenario. Please note that, even fixing the scaling values to S^w, the number of networks increases factorially with the number of queues in the CPS. Nevertheless, depending on the particular goal that is pursued, the formulation of the method is such that it can be easily simplified for deriving practical results. See [15] for an example.

5.1 Setting the scaling values

In order to reduce the complexity of the method, we present a first set of scaling values S^w that respects Theorem 3.2. Please note that S^w is not the only possible choice for the scalers of the upper bounding networks.

THEOREM 5.1. *It is given a CPS and a network (**n**,S^w), where, for any stage j and for any $p \in [1, ..., j-1]$, S^w is:*
$$S^w_{j,p} = \frac{R^{up}_{n_j} - R^{p-up}_{n_j}}{R^{up}_{n_p}}, \tag{9}$$
*with $R^{p-up}_{n_j}$ the service rate at the CPS node n_j when the set of CPS nodes $[n_p, ..., n_N]$ is active. Then (**n**,S^w) satisfies Theorem 3.2 and it is therefore an upper bounding network for the CPS.*

For the proof see Section 8.4. Let us show what is the intuition behind the choice of scaling values in this result. Let us consider stage j and a state vector $\mathbf{Y}$ for the upper bounding network, whose first GPS active node for the fresh traffic is at stage $p < j$. Whatever is the subset of active queues for the fresh traffic at stages $[p, ..., j-1]$, this choice of scaling values makes the traffic coming from stage p alone reduce the service rate at $R^{p-up}_{n_j}$, i.e., as all the queues for the fresh traffic at stages $[p, ..., j-1]$ were active. The effect of this approximation is mitigated by the fact that in our method we use different upper bounding networks, associated to different assignments of CPS nodes to stages.

5.2 Description of the scenario

In order to evaluate the performance of the introduced method in terms of quality of the achieved bounds, on the stability region and on the per-packet delay, we introduce a simple, but realistic, example where the CPS model applies. We consider a scenario with Device-to-Device (D2D) transmission-receiver pairs [1]. By means of D2D communications, devices transmit to the intended receiver, typically in proximity, by direct communication, i.e., without passing through a central entity. Depending on the implementation, D2D communications can also reduce the role of the central entity in terms of synchronization and management by self organizing transmissions [16]. In the case under analysis, we consider the D2D transmission pairs able to sense surrounding interference and choose the best Modulation Coding Scheme (MCS) that allows decoding at the receiver. In practice, we use a Full Frequency Reuse scheduling approach, i.e., transmitters exploit the whole available bandwidth when they have traffic to serve and simultaneous transmissions are enabled by means of the correct selection of MCSs. We assume a static scenario, where devices do not move and channel characteristics do not change over time. If so, the interference sensed at the receiver is univocally determined by the set of active transmitters in the system. Due to the fact that MCSs are chosen depending on interference, the throughput achieved by each transmitter, a.k.a, the service rate of its transmission queue, is univocally determined by the set of active transmitters at any time t. Therefore, the presented scenario can be studied by a CPS model.

5.3 Stability Region Evaluation

In the following section we compare the analytical sufficient condition for stability with the set of achievable throughput we got from the simulations of the presented wireless environment. We evaluate the particular scenario depicted in Fig. 2, where 3 D2D pairs are present. Larger scenarios can be easily studied, though.

Table 1 summarizes the values of the variables used during the simulations. We use a Free Space Path Loss model. Incoming traffic at

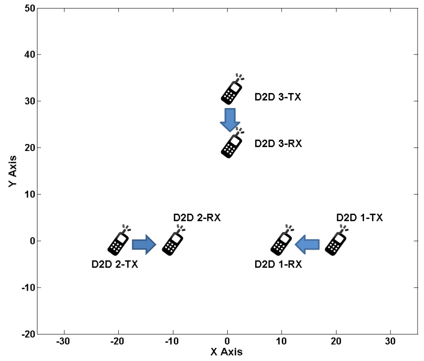


Figure 2: Scenario under analysis

Table 1: Simulation Setup

Carrier	$2.4\ GHz$
Transm. Bandwidth	$20\ MHz$
Distance between D2D TX/RX	$10\ m$
Distance D2D RX Interfere TX	$\sim 30\ m$
$\mathcal{N}$	$3.98 * 10^{-18}\ W/Hz$
P_{TX}	200 mW
max. packet length	12,000 bits
max. transmission rate	$90\ Mb/s$

the transmission queues was following a leaky bucket characterization, with fixed burstiness (12000 bits, i.e., a packet) and long term rates varying from zero to the maximum allowed transmission rate (90 Mb/s).

The analytical representation of the sufficient conditions for stability of the presented scenario can be achieved directly from Theorem 3.3. In particular, analysing network ($\mathbf{n}$,S^w), sufficient conditions for stability are:

$$\begin{cases} \rho_{n_1} \leq R^{sat}_{n_1}, \\ \rho_{n_2} \leq R^{up}_{n_2} - \frac{R^{up}_{n_2} - R^{sat}_{n_2}}{R^{sat}_{n_1}} \rho_{n_1}, \\ \rho_{n_3} \leq \max(R^{sat}_{n_3}, R^{up}_{n_3} - \frac{R^{u}_{n_3}p - R^{sat}_{n_3}}{R^{sat}_{n_1}} \rho_{n_1} - \frac{R^{up}_{n_3} - R^{2-up}_{n_3}}{R^{up}_{n_2}} \rho_{n_2}). \end{cases} \tag{10}$$

The union of the sufficient conditions for stability achieved from any of the possible sorting $\mathbf{n}$ is the analytical representation of the sufficient conditions for stability obtained through the method introduced in Section 3.

In Fig. 3 we show both the sufficient conditions for stability achieved through (10), both the set of achievable throughputs obtained through simulations, i.e., by means of Montecarlo experiments. Fig. 4 represents instead the difference among the two, taking as reference the maximum achievable rate of D2D-TX 3. As expected, the sufficient conditions for stability underestimates the set of achievable rate obtained through simulation. The maximum underestimation of the achievable rate D2D-TX 3 is close to 25 Mb/s. By the way, such underestimation is registered in very small parts of the stability region. Indeed, the average underestimation achieved analytically is as low as $5.92\ Mb/sec$, when the average achieved by D2D-TX 3 is $41.25\ Mb/s$. All in all, even if achieved by means of worst case analysis, the sufficient condition for stability are really close to the set of achievable rates obtained through simulation.

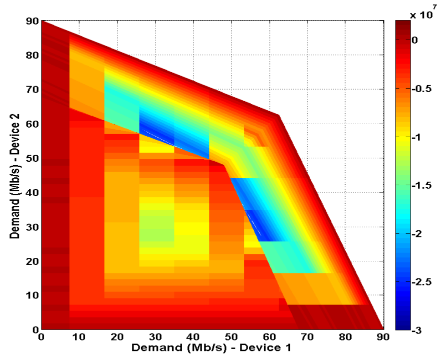


Figure 4: Error Maximum Achievable Rate (Mb/s), D2D-TX 3.

5.4 Evaluation of Delay Bounds

In order to evaluate the delay bounds presented in Section 3.2, for each of the simulations used in Section 5.3, we computed the per-packet simulated delay. In case the arrival rates used in the simulation were a set of stable arrivals from (10), we compared the per-packet simulated delay with the delay bound achieved through our methodology. In particular, for each D2D pair, we evaluated the ratio among the worst per-packet simulated delay and the its bound. The average ratio among the worst per-packet simulated delay and the bound computed as in Section 5.3 has been of the $88.95\% \pm 2.64\%$ (95% confidence interval). The minimum registered ratio was 66.41%, while the maximum ratio has been 99.69%. As expected, the bound on the delay has never been violated. Fig. 5 presents the delay distribution of the three D2D pairs per-packet delay when the difference among the bound and the simulation was the greatest.

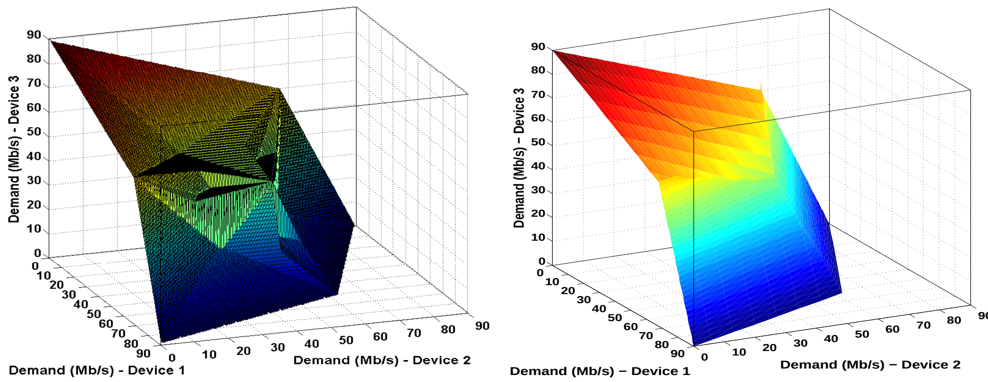
As it is clear from the simulations performed, the bound on the per-packet delay is close to the one a CPS server could experience, even though we analysed the system through the use of upper bounding networks.

6. CONCLUSIONS

In the present paper, we propose a new method for the analysis of CPSs, valid for any number of nodes, and we describe a technique for deriving practical results, which exploits the characteristics of the considered system. Our results show that even on a worst case framework as Network Calculus, taking into account the dynamics of the system through a characterization of the input traffic brings to substantially better resource allocations than those obtainable through a static model of the system. We plan to extend this work in two main directions. On one side, we plan to extend our framework to include CPS where nodes are interconnected, and to consider dependency on queue length rather than only on queue occupancy. On the other hand, we plan of applying our method to interference limited wireless scenarios, to multiprocessor systems, in data centers, and in general on all those CPS systems for which simulation has been so far the main method of analysis, in order to get some insight on their performance.

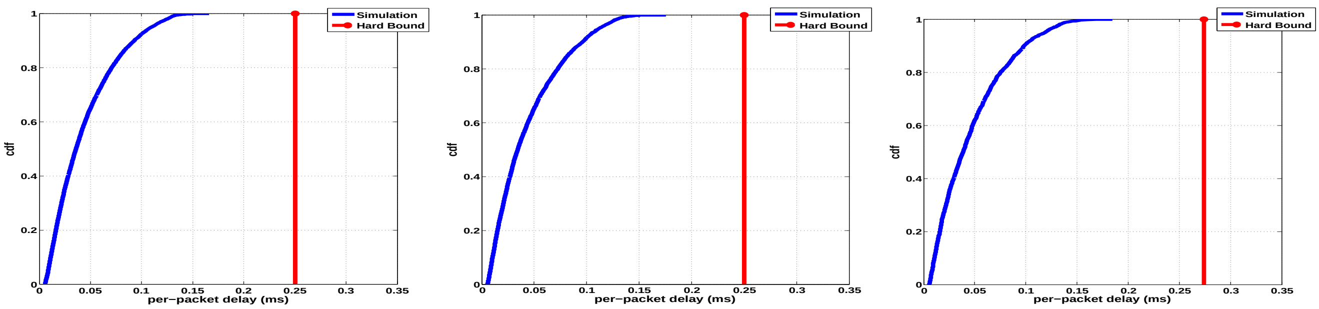
7. REFERENCES

[1] A. Asadi and V. Mancuso. A survey on opportunistic scheduling in wireless communications. *IEEE Communications Surveys & Tutorials*, 15(4):1671–1688, 2013.



(a) Inner Bound Stability Region (b) Achievable Throughput Simulation

Figure 3: Maximum Achievable Throughput, Analysis vs. Simulations.



(a) D2D TX1 (b) D2D TX2 (c) D2D TX3

Figure 5: Per-packet delay (cdf). Simulations vs. Bound

[2] Thomas Bonald, Sem C. Borst, Nidhi Hegde, and Alexandre Proutiére. Wireless data performance in multi-cell scenarios. In *SIGMETRICS*, pages 378–380, 2004.

[3] Sem Borst, Nidhi Hegde, and Alexandre Proutiere. Interacting queues with server selection and coordinated scheduling: application to cellular data networks. *Annals of Operations Research*, 170(1):59–78, 2009.

[4] Sem C. Borst, Onno J. Boxma, and Predrag R. Jelenkovic. Coupled processors with regularly varying service times. In *INFOCOM*, pages 157–164, 2000.

[5] Sem C. Borst, Matthieu Jonckheere, and Lasse Leskelä. Stability of parallel queueing systems with coupled service rates. *Discrete Event Dynamic Systems*, 18(4):447–472, 2008.

[6] Jean-Yves Le Boudec and Patrick Thiran. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*, volume 2050 of *LNCS*. Springer, 2001.

[7] Florin Ciucu. *Scaling properties in the stochastic network calculus*. PhD thesis, University of Virginia, Charlottesville, VA, USA, 2007.

[8] Guy Fayolle and R. Iasnogorodski. Solutions of functional equations arising in the analysis of two server queueing models. In *Performance*, pages 289–303, 1979.

[9] Markus Fidler. Survey of deterministic and stochastic service curve models in the network calculus. *IEEE Communications Surveys and Tutorials*, 12(1):59–86, 2010.

[10] Markus Fidler and Jens B. Schmitt. On the way to a distributed systems calculus: an end-to-end network calculus with data scaling. *SIGMETRICS*, 34(1):287–298, jun 2006.

[11] F. Guillemin and D. Pinchon. Analysis of the weighted fair queuing system with two classes of customers with exponential service times. In *Journal of Applied Probability*, 2004.

[12] Matthieu Jonckheere and Sem C. Borst. Stability of multi-class queueing systems with state-dependent service rates. In *VALUETOOLS*, page 15, 2006.

[13] Ritesh Madan, Jaber Borran, Ashwin Sampath, Naga Bhushan, Aamod Khandekar, and Tingfang Ji. Cell association and interference coordination in heterogeneous lte-a cellular networks. *IEEE JSAC*, 28(9):1479–1489, 2010.

[14] B. Rengarajan, C. Caramanis, and G. de Veciana. Analyzing queuing systems with coupled processors through semidefinite programming. 2008.

[15] Christian Vitale, Vincenzo Mancuso, and Gianluca Rizzo. *Modelling D2D Communications in Cellular Access Networks via Coupled Processors*, 2014. http://publications.hevs.ch/index.php/attachments/single/715.

[16] Xinzhou Wu, Saurabha Tavildar, Sanjay Shakkottai, Tom Richardson, Junyi Li, Rajiv Laroia, and Aleksandar Jovicic. Flashlinq: A synchronous distributed scheduler for peer-to-peer ad hoc networks. *IEEE/ACM Transactions on Networking (TON)*, 21(4):1215–1228, 2013.

8. APPENDIX

8.1 Proof of Theorem 3.2

If (3) is substitute into (1), we get that:

$$O_j(\mathbf{Y}(t)) \le R^{up}_{n_j}(\mathbf{Y}(t)) \le R_{n_j}(\mathbf{Y}(t)) \tag{11}$$

Here $R_{n_j}(\mathbf{Y}(t))$ is the service rate of queue n_j of the CPS when the queues active at the CPS at time t are *only* all those which correspond to active queues at the network at time t. (11) ensures that the service rate of each queue of the network is always inferior to the one of the correspondent queues in the CPS, when all active queues in the network are associated to active queues at the CPS.

We now prove that such property is sufficient for the network to respect Theorem 3.1. That is, we prove that if (11) holds at any node and any time $t \geq 0$, then $O_j(\mathbf{Y}(t)) \leq R_{n_j}(\mathbf{I}(t))$ for any node j and any time $t \geq 0$. We prove by induction. Given arrivals are packetized, we assume time is discrete, and we consider the sequence of relevant network events, given by packet arrivals and departures at a node (we consider events taking place both at the CPS and at the network).

Let us consider the general case in which queues are nonempty at $t = 0$. We assume at each CPS queue backlog for fresh arrivals is the same, at $t = 0$ as at its corresponding queue in the network. Let us consider the first event of packet departure, at t_1 and let us assume that it takes place at a node in the network. Until the first event of packet departure, $O_j(\mathbf{Y}(t)) \leq R_{n_j}(\mathbf{I}(t))$ holds for any node j. Indeed, as arrivals are the same at every time instant at each queue and at its corresponding queue in the network, until t_1 there has been no queue which is empty at the CPS and active at the network, or vice versa, so that $\forall\ j,\ \forall\ t \in [0, t_1],\ R_{n_j}(\mathbf{Y}(t)) = R_{n_j}(\mathbf{I}(t))$. The first event of packet departure cannot be at at a node of the network, otherwise this would mean that that packet has been served faster at that node, at some time $t < t_1$. This completes the first step of induction.
For the k-th step, we prove it by contradiction. We assume the network satisfies Eq. (11), until time t^*, with $t_k \leq t^* \leq t_{k+1}$, where t_k, t_{k+1} are the times of the last relevant network event before t^*, and of the first after t^*, respectively. So we assume at t^* at least one of the queues dedicated to the fresh sources in the network is served faster than the corresponding node of the CPS, i.e., Theorem 3.1. That is, there exist a node j such that $O_j(\mathbf{Y}(t^*)) > R_{n_j}(\mathbf{I}(t^*))$. Because of Eq. (11), $O_j(\mathbf{Y}(t^*)) \leq R_{n_j}(\mathbf{Y}(t^*))$. As the CPS is monotonic, this implies that at time t^* at least one of the active queues in the CPS is associated to an empty queue in the network. Due to the fact that the arrivals are exactly the same at all CPS queues and at the mapped queues of the network, the queue that is empty in the network and not in the CPS should have been served faster than at the CPS, at least for an instant, in a previous moment of t^*, which is in contradiction with what assumed.

8.2 Proof of Theorem 3.3

As the considered upper bounding network associated to the couple (**n**,**S**) and satisfying Theorem 3.2 is feed forward, we analyze it stage by stage. At each stage, we compute the service curve of the GPS node for fresh traffic. In what follows, we show that if stages 1 to $j-1$ are stable, the service curve for fresh traffic at the j-th stage, $j = 1, ..., N$ is:

$$\beta_{n_j}(t) = max\left(R^{sat}_{n_j}t, \left[\left(R^{up}_{n_j} - \sum_{p=1}^{j-1} S_{j,p}\rho_{n_p}\right)t - T_{n_j}\right]^+\right) \tag{12}$$

where

$$T_{n_j} = \frac{\sum_{p=1}^{j-1} S_{j,p}\sigma^*_{n_p}}{R^{up}_{n_j} - \sum_{p=1}^{j-1} S_{j,p}\rho_{n_p}}$$

$$\sigma^*_{n_p} = \begin{cases} \sigma_{n_p}, & \text{if } \rho_{n_p} \leq R^{sat}_{n_p}, \\ \sigma_{n_p} + \frac{(\rho_{n_p} - R^{sat}_{n_p})(\sum_{k=1}^{p-1} S_{p,k}\sigma^*_{n_k})}{R^{up}_{n_p} - R^{sat}_{n_p} - \sum_{k=1}^{p-1} S_{p,k}\rho_{n_k}} & \text{otherwise.} \end{cases} \tag{13}$$

Note that $[x]^+$ stands for $\max(x, 0)$. We show that these expressions hold by induction.
At first stage the service curve is $\beta_{n_1}(t) = R^{sat}_{n_1}t$. If the node is stable i.e. if $\rho_{n_1} \leq R^{sat}_{n_1}$, a valid arrival curve for the traffic going from stage 1 to stage 2 (and to each of the other stages) is $\rho_{n_1}t + \sigma_{n_1}$. Note that for simplicity we do not use in the output characterization the link limit of the serving node. Then after passing the scaler and the policer, an arrival curve for such traffic at the input of the GPS node of the second stage, is

$$\gamma_{n_2} = min\left((R^{up}_{n_2} - R^{sat}_{n_2})t, S_{2,1}\rho_{n_1}t + S_{2,1}\sigma_{n_1}\right).$$

The service curve offered by the GPS node to this traffic is $(R^{up}_{n_2} - R^{sat}_{n_2})t$. Therefore the arrival curve for this traffic at the output of the second stage is the same as at its input. By exploiting the well known formula for the leftover service curve at a GPS node [9], a service curve for fresh traffic at the second stage is:

$$\beta_{n_2}(t) = max\left(R^{sat}_{n_2}t, [(R^{up}_{n_2} - S_{2,1}\rho_{n_1})t + T_{n_1}]^+\right),$$

where:

$$T_{n_1} = \frac{S_{2,1}\sigma_{n_1}}{R^{up}_{n_2} - S_{2,1}\rho_{n_1}},$$

The node is stable (and the backlog at the queue for fresh traffic bounded) if $\rho_{n_2} \leq max(R^{sat}_{n_2}, R^{up}_{n_2} - S_{2,1}\rho_{n_1})$,.u.e. if the fresh flow rate is inferior than the service rate of the leftover service curve. An arrival curve for the traffic going from the second stage to each of the following ones is given by

$$\gamma'_{n_2}(t) = (\gamma_{n_2} \oslash \beta_{n_2})(t) = \rho_{n_2}t + \sigma^*_{n_2},$$

where γ_{n_2} is the arrival curve of the fresh traffic assigned to the second stage, i.e. the leaky bucket characterization given by the couple $\{\rho_{n_2}, \sigma_{n_2}\}$. $\oslash$ is the min-plus deconvolution operator [6]. $\sigma^*_{n_2}$ is given by:

$$\sigma^*_{n_2} = \begin{cases} \sigma_{n_2} & \text{if } \rho_{n_2} \leq R^{sat}_{n_2} \\ \sigma_{n_2} + \frac{(\rho_{n_2} - R^{sat}_{n_2})S_{2,1}\sigma_{n_1}}{R^{up}_{n_2} - R^{sat}_{n_2} - S_{2,1}\rho_{n_1}} & \text{otherwise} \end{cases}$$

Let us now assume that (12) holds up to stage $j-1$ and that all these stages are stable. Then, a valid arrival curve for the flow coming from the generic upper stage $k \in [1, ..., j-1]$ is given by:

$$\gamma'_k(t) = (\gamma_{n_k} \oslash \beta_{n_k})(t) = \rho_{n_k}t + \sigma^*_{n_k},$$

where $\sigma^*_{n_k}$ respects (13). An arrival curve for the traffic entering the GPS node at stage j coming from upper stages, at the output of the policer, $\gamma^{IN}_j(t)$, is:

$$\gamma^{IN}_j(t) = min\left((R^{up}_{n_j} - R^{sat}_{n_j})t, \sum_{k=1}^{j-1} S_{j,k}\rho_{n_k}t + \sum_{k=1}^{j-1} S_{j,k}\sigma^*_{n_k}\right).$$

We note that $\gamma^{IN}_j(t)$ is also an arrival curve for the same traffic at the node. Using again the result for the leftover service curve, the service curve for the traffic coming from the outside source at j-th stage satisfies (12). Again, by imposing the node stability condition we get $\rho_{n_j} \leq max(R^{sat}_{n_j}, R^{up}_{n_j} - \sum_{p=1}^{j-1} S_{j,p}\rho_{n_p})$.

8.3 Proof of Theorem 3.4

The proof follows the same line as the one of Theorem 3.3. Once the service curve for fresh flow is derived, backlog and delay bounds for fresh traffic are derived as bounds to the maximum vertical and horizontal distance between arrival curves and service curves, exploiting some basic Network Calculus results [6]. We already noted that backlog bounds for queues in the network are also valid bounds for the corresponding CPS queues. We now prove that the same holds also for delay bounds. Let us compute the delay of a generic bit entering at time t its queue n_j at the CPS and at the j th queue for fresh traffic. Respectively, $q_{n_j}(t)$ and $q_j(t)$ are the backlog at those queues at time t. The delay at the CPS is:

$$D_{CPS} = t' - t, \tag{14}$$

where t' is the moment in which

$$\int_t^{t'} R_{n_j}(x)\,\mathrm{d}x = q_{n_j}(t) + 1.$$

The delay at its corresponding queue j is instead:

$$D_j = t'' - t, \tag{15}$$

where t'' is the moment in which

$$\int_t^{t''} R_j(x)\,\mathrm{d}x = q_j(t) + 1.$$

Considering that $R_j(t) \leq R_{n_j}(t)$ at any time t, then:

$$\int_A^B R_{n_j}(x)\,\mathrm{d}x \leq \int_A^B R_{n_j}(x)\,\mathrm{d}x \tag{16}$$

for any interval $[A, B]$ chosen. Then, due to the fact that $q_j(t) \geq q_{n_j}(t)$, it directly follows that $t'' \geq t'$, proving the theorem.

8.4 Proof of Theorem 5.1

Let us assume a generic system state $\mathbf{Y}$ for the network introduced and k be the first stage whose $(\mathbf{Y})_k = 1$. Substituting (9) into (3) we can obtain:

$$\sum_{p=1}^{j-1} O_p(t) S_{j,p} \geq O_k(t) S_{j,k} = R_{n_j}^{up} - R_{n_j}^{k-up}, \tag{17}$$

that satisfies the upper bounding condition in Theorem 3.2.