A product-form model for the performance evaluation of a bandwidth allocation strategy in WSNs

ANDREA MARIN, Università Ca' Foscari Venezia
and SABINA ROSSI, Università Ca' Foscari Venezia
and DARIO BURATO, Università Ca' Foscari Venezia
and ANDREA SINA, Università Ca' Foscari Venezia
and MATTEO SOTTANA, Università Ca' Foscari Venezia

Wireless Sensor Networks (WSNs) are important examples of Collective Adaptive System (CAS) which consist of a set of motes that are spatially distributed in an indoor or outdoor space. Each mote monitors its surrounding conditions, such as humidity, intensity of light, temperature, vibrations but also collects complex information such as images or small videos and cooperates with the whole set of motes forming the WSN in order to allow the routing process. The traffic in the WSN consists of packets which contain the data harvested by the motes and can be classified according to the type of information that they carry. One pivotal problem in WSNs is the bandwidth allocation among the motes. The problem is known to be challenging due to the reduced computational capacity of the motes, their energy consumption constraints and the fully decentralised network architecture. In this paper we study a novel algorithm to allocate the WSN bandwidth among the motes by taking into account the type of traffic they aim to send. Under the assumption of a mesh network and Poisson distributed harvested packets, we propose an analytical model for its performance evaluation that allows a designer to study the optimal configuration parameters. Although the Markov chain underlying the model is not reversible, we show it to be ρ-reversible under a certain renaming of states. By an extensive set of simulations, we show that the analytical model accurately approximates the performance of networks that do not satisfy the assumptions. The algorithm is studied with respect to the achieved throughput and fairness. We show that it provides a good approximation of the max-min fairness requirements.

Categories and Subject Descriptors: C.2.1 [**Computer-communication Networks**]: Network Architecture Design—*Distributed Networks*; C.2.1 [**Computer-communication Networks**]: Network Architecture Design—*Wireless Networks*; C.4 [**Performance of Systems**]: —*Modelling techniques*

Additional Key Words and Phrases: Wireless sensor networks, bandwidth allocation, Markov models, product-forms

1. INTRODUCTION

Wireless Sensor Networks (WSNs) are employed to collect data from an environment and send them to monitoring and control applications. A large set of motes equipped with sensors and a wireless antenna are deployed in an environment in order to sense and measure a large variety of physical phenomena such as temperature, humidity, pressure, radiation, air pollution levels, noise level but also more sophisticated events such as the number of people in an area or the movement of an object. Practical applications include measuring tremors in seismic areas [Lopes Pereira et al. 2014], monitoring the concentration of polluting agents [Yi et al. 2015] and collecting data from a large environment for scientific purposes [Mainwaring et al. 2002]. The motes forming a WSN are typically simple systems powered by batteries with a limited memory and computational power. Many research efforts are devoted to develop new technologies or protocols that can extend the lifetime of a network, such as the capability of nodes of harvesting energy from the environment and hence extend their life (see, e.g., [Gelenbe and Marin 2015; Tan and Tang 2017; Tang and Tan 2017; Tunc and Akar 2017] for a survey of the state of the art of the technology and the available models). Although in this paper we will mainly focus on the bandwidth allocation problem, the solution that we propose can be also used for balancing the energy consumption of the nodes forming a network.

Nodes in a WSN are equipped with a wi-fi antenna that allows for omnidirectional communications. Despite their simple structure, the motes of WSNs must coordinate themselves in order to implement some sort of energy efficient routing protocol whose goal is to deliver data harvested by an arbitrary sensor to one of the base stations. These are special nodes that collect the data harvested by the motes' sensors and may govern the actuators (e.g., they may raise a fire alarm in case some motes communicate a sudden rise in the temperature). In this setting, the wireless channel represents a shared resource among the motes whose access must be carefully governed in order to avoid the congestion collapse of the WSN (see, e.g., [Akyildiz and Kasimoglu 2004; Sankarasubramabiam et al. 2003; Cherian and Nair 2014; Chitnis et al. 2009; Khan et al. 2012]). In this view, we can say that large WSNs represent an example of Collective Adaptive System (CAS) [Frescha 2015; Feng et al. 2016] since they share a decentralised control, an autonomous behaviour of the motes and a competition of the bandwidth resource.

The structure of the communication network formed by the motes may be based on several architectures: all the nodes may implement the same wireless transmission protocol in order to form a decentralised ad-hoc network or there may be a hierarchical structure as in the cluster-tree topology [Akyildiz and Kasimoglu 2004; Hanzalek and Jurĉik 2010].

It is often the case that the communication infrastructure formed by the sensors is used to transmit different types of data which may have different importance and priority. For instance in monitoring an environment the data associated with sensing vibrations may have higher priority than those measuring other phenomena such as the humidity since the former may indicate an earthquake or a tremor. Therefore, it appears natural that in the development of WSN protocols one should consider the assignment of the resources (frequency spectrum or bandwidth) according to the priorities of the sensed data. On the other hand, due to the limited amount of computational power and the limited energy supplies of the sensors, the design of protocols for the resource assignment should be as simple as possible and avoid energy loss due to heavy computations or packet collisions (see, e.g., [Tan et al. 2015; Tan and Wu 2016]).

In this paper we address the problem of the bandwidth allocation among traffic classes with different priority by modelling a stateless protocol inspired by *back-CHOKe* [Pan et al. 2000] and by studying its performance. The main idea of the proposed allocation control scheme, named *Fair Allocation Control Window* (FACW), is that each sensor maintains a control window of size N that stores the traffic classes of the latest packets that have been sent by its neighbours (i.e., the other sensors it can listen to) or by itself. Each traffic class c can be present in the window at most h_c times, where higher values of h_c correspond to higher traffic priority. When a sensor collects data with type c, it behaves as follows: if the number of c-classes in its window is less than h_c then it transmits the packet, otherwise it waits a random time (or drops the packet) and retries later. If the packet is sent then the window is updated by inserting an object of class c according to a First-In-First-Out (FIFO) policy. We study the behaviour of this admission control algorithm under different scenarios.

The main contribution of the paper is the introduction of a Markov model that, under certain assumptions, describes the evolution of the window states on time. The two main assumptions required for the exact stationary analysis of the Markov chain are: (i) the packets are harvested according to independent time-homogeneous Poisson processes and (ii) the motes form a mesh network, i.e., a network where every node is in the transmission range of the others. We study the robustness of the performance indices derived analytically by comparing them with the estimates obtained by stochastic simulations of models that violate these assumption. Interestingly, we show that the model is ρ-reversible [Whittle 1986; Kelly 1979; Marin and Rossi 2014b; Marin and Rossi 2014a], although not reversible. This means that the time-reversed Markov chain is stochastically identical to the original one modulo a renaming of the states. Thanks to this property we can give a closed form expression for its invariant measure. The stationary performance indices are expressed as functions of the normalising constant which is derived algorithmically according to a new convolution algorithm. The availability of a numerically efficient approach for the performance evaluation allows for the parameterisation of a WSN without resorting to computationally expensive simulations. Finally, we compare the performance indices given by model with those obtained by simulation in WSNs which do not satisfy the model's assumptions. We show that the model gives accurate predictions also in these cases and hence it is useful for studying and optimising large-scale WSNs.

This work is an extension of the paper proposed in [Marin and Rossi 2016b]. With respect to the conference version, we have included here all the proofs of the theorems and propositions, and the analysis of robustness of the model by resorting to the stochastic simulation. *Structure of the paper*. In Section 2 we describe our bandwidth allocation control scheme. In Section 3 we present the performance evaluation of FACW and give the algorithm for the computation of the performance indices. Section 4 shows our scheme at work under various scenarios. In Section 5 we test the robustness of the model with respect to the stochastic simulations of WSNs which do not form a mesh network. Section 6 discusses some related work. Finally, Section 7 concludes the paper.

2. THE FACW ALGORITHM

In this section we first introduce the design goals of the Fair Allocation Control Window (FACW) algorithm and then we describe how it works.

2.1 Design goals

The design of our fair allocation control algorithm aims at satisfying the following goals:

(1) *Stateless architecture.* Due to the limited physical resources in the motes, we aim at reducing the computational cost and the memory usage at each mote.
(2) *Localized behaviour.* The algorithm decisions are based on local information and do not affect the whole system.
(3) *Avoid transmissions of extra packets.* Control packet transmission should be avoided in order to reduce the energy dissipation of the system.
(4) *Fair bandwidth allocation among different traffic flows with the same priority.* Our algorithm aims at satisfying the *max-min* fairness criterion among different traffic flows with the same priority. We will discuss this objective in more details in Section 3, but intuitively we do not want that a flow with low requirements is slowed down while there exists another flow with the same priority which is using more resources [Hahne 1991; Bertsekas and Gallager 1992].
(5) *Flexible regulation of traffic priorities.* The allocation of the bandwidth follows a soft-priority based scheme in order to prevent lower priority traffic flows to starve because of the presence of a greedy higher traffic flow.
(6) *Easiness of implementation.* The algorithm must be easy to implement into a proper protocol within the actual motes software.

2.2 The Fair Allocation Control Window (FACW) algorithm

The main idea of FACW is that data traffic in a WSN can be classified into a finite set of M classes $\mathcal{K} = \{c_1, c_2, \ldots, c_M\}$. Each mote maintains a control window of size N in which the classes of the latest N transmissions (listened or performed) are stored. In the window, at most h_c entries of class c can appear, where $1 \leq h_c \leq N$. We stress the fact that the window stores only the class identifier of a transmission and not the sent packet. So, if we assume a practical situation with 16 classes, each class

can be encoded by 4 bits and hence a window can be stored in few bytes. In case the mote generates a packet of class c when in its window there are already h_c entries of class c, the packet is rescheduled for transmission after a back-off time or is simply dropped. Otherwise, in case of generation of a class c packet and the number of c-entries in the window is strictly lower than h_c, then the packet is sent and the window is updated according to a FIFO policy. It should be clear that larger window sizes imply a lower bandwidth usage, whereas lower values of N make the transmission more aggressive. The role of h_c is that of modelling class priority. Allowing more entries of a class c in the control window reduces the probability of c-packet dropping/delaying and hence its priority is larger than that of a traffic class d with $h_d < h_c$. The initialisation of the window is arbitrary. If necessary, we can assume that there exists a class c of data traffic (e.g., the packets used for controlling the routing) that has $h_c = N$ and whose rate is slow. The presence of this class ensures that the starvation of all the other traffic classes never occurs because of the control window.
In the following sections we present a numerically tractable model that can be used to parametrise the protocol, i.e., decide the window size and the values of h_c for $c \in \mathcal{K}$.

3. ANALYTICAL MODEL FOR MESH NETWORKS

In this section we present a stochastic model based on Continuous Time Markov Chains (CTMC) for the FACW allocation scheme. The numerical tractability of this model allows us to use it to set the parameters in the protocol implementation, i.e., the window size N and the values for the threshold h_c for each traffic class $c \in \mathcal{K}$. The model considers a single window and is subject to the following assumptions:

—Packets are generated according to independent Poisson processes whose rates may depend on the window state. This allows us to model situations in which the mote modulates its harvesting rate according to the population of the control window. We can also deal with the case in which the class c packets which are not sent are delayed and hence the packet generation rate is increased because the sensor data production rate is summed to the packet retransmission rate.
—We consider a network topology in which every mote senses the transmission of every other mote. This is a common assumption in tree-structured WSNs in which it is assumed that all the motes with the same parent interfere in their transmissions because they are relatively geographically close. This requirement is needed because here we aim to study the network performance and hence we assume that all the nodes share the same contention window. Nevertheless, if we are interested in the analysis of the performance of a single mote, this requirement is not needed.

3.1 The stochastic model

We consider a set $\mathcal{K} = \{c_1, c_2, \dots, c_M\}$ of M distinct traffic classes and assume that each node maintains a window $\mathcal{W}$ of size N storing the transmission classes of the most recent sensed data according to a FIFO policy. An arrival can be due to a sensor data harvesting or to a listening to another node transmission. We denote the state of the window by $\vec{x} = (x_1, x_2, \dots, x_N)$, where $x_i \in \mathcal{K}$, and let $|\vec{x}|_c = \sum_{i=1}^{N} \delta_{x_i = c}$ be the total number of occurrences of class c in $\mathcal{W}$. We assume that data of different traffic classes are generated according to independent Poisson processes whose rates $\lambda_c(j)$, with $c \in \mathcal{K}$ and $1 \le j \le N$, depend on the number of objects $j = |\vec{x}|_c$ of class c that are present in the window. Clearly, the process $X(t)$ that describes the state of $\mathcal{W}$ is a homogeneous continuous time Markov chain (CTMC) with finite state space. In the window there can be at most h_c objects of class c, with $c \in \mathcal{K}$. If $h_c = N$ then there is no constraint on the maximum number of objects of the same class in the window. Let $\vec{x} = (x_1, \dots, x_N)$ be the state of the control window, then the transition rates in the CTMC infinitesimal generator are: for $\vec{x} \neq \vec{x}'$,

$$q(\vec{x}, \vec{x}') = \begin{cases} \lambda_c(|\vec{x}|_c) & \text{if } \vec{x}' = (c, x_1, \dots, x_{N-1}) \text{ and } |\vec{x}|_c < h_c \\ 0 & \text{otherwise.} \end{cases}$$

3.2 Closed form stationary distribution

We derive the stationary distribution of process $X(t)$. The state space of $X(t)$ is $\mathcal{S} = \{\vec{x} \in \mathcal{K}^N : |\vec{x}|_c \leq h_c \text{ for all } c \in \mathcal{K}\}$. Note that the state space of $X(t)$ is finite and its transition graph is irreducible. Hence the CTMC has a unique limiting distribution independent of its initial state.

THEOREM 3.1. *The stationary distribution $\pi(\vec{x})$ of $X(t)$ for the FIFO policy is given by the following expression:*

$$\pi(\vec{x}) = \frac{1}{G} \prod_{c \in \mathcal{K}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j) \,, \tag{1}$$

where $G = \sum_{\vec{x} \in \mathcal{S}} \prod_{c \in \mathcal{K}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)$.

The proof of Theorem 3.1 is based on the notion of *ϱ-reversibility* [Marin and Rossi 2014b; Marin and Rossi 2014a; Marin and Rossi 2016a] which generalises the concepts of reversibility [Kelly 1979] and dynamic reversibility [Whittle 1986] by considering those CTMCs which are stochastically identical to their reversed process modulo a state renaming ϱ. More formally, a CTMC $X(t)$ is reversible if it is stochastically identical to its reversed process, it is dynamically reversible if it is stochastically identical to its reversed process where the state names are changed according to an involution ϱ over the state space of $X(t)$, whereas it is ρ-reversible when ϱ is a generic state renaming. Any ϱ-reversible Markov chain is characterized by a set of detailed balance equations expressed in terms of the steady-state distribution π and the transition rates q_{ij}, for $i, j \in \mathcal{S}$, of the Markov process.

PROPOSITION 3.2. *[Marin and Rossi 2016a] A stationary CTMC with state space $\mathcal{S}$ is ϱ-reversible with respect to a renaming ϱ on $\mathcal{S}$ if and only if there exists a set of positive real numbers π_i summing to unity, with $i \in \mathcal{S}$, such that the following system of detailed balance equations are satisfied: for $i, j \in \mathcal{S}$, $i \neq j$:*

$$\pi_i q_{ij} = \pi_j q_{\varrho(j)\varrho(i)}$$

and $q_i = q_{\varrho(i)}$, where $q_i = \sum_{j \neq i} q_{ij}$. If such a solution π_i exists then it is the stationary distribution of $X(t)$.

The steady-state distribution of a ϱ-reversible CTMC can be expressed in terms of the transition rates.

PROPOSITION 3.3. *[Marin and Rossi 2016a] Let $X(t)$ be a stationary CTMC with state space $\mathcal{S}$ which is ϱ-reversible with respect to a renaming ϱ over $\mathcal{S}$. Let $i_0 \in \mathcal{S}$ be and arbitrary reference state. Let $i \in \mathcal{S}$ and $i = i_n \to i_{n-1} \to \cdots \to i_1 \to i_0$ be a chain of one-step transitions. Then, for $C_{i_0} \in \mathbb{R}^+$,*

$$\pi_i = C_{i_0} \prod_{k=1}^{n} \frac{q_{\varrho(i_{k-1})\varrho(i_k)}}{q_{i_k i_{k-1}}} \,. \tag{2}$$

The proof of Theorem 3.1 is structured as follows: we first make a claim that $X(t)$ is ϱ-reversible and then, from Proposition 3.3, we derive Expression (1) of the stationary distribution. Finally, by using Proposition 3.2 we prove the claim. The detailed proof of the theorem is given in the Appendix.

In most practical applications, we are not interested in knowing the stationary probability of observing a state of $X(t)$ since the transmission classes are temporally ordered in it. More often, we are interested in knowing the stationary probability of observing a state in which the occurrences of each class $c_1, \ldots, c_M$ are $n_{c_1}, \ldots, n_{c_M}$ whatever is their order. Corollary 3.4 provides an analytical expression for such an aggregated equilibrium probability. The proof is given in the Appendix.

COROLLARY 3.4. *Let $\mathbf{n} = (n_{c_1}, \ldots, n_{c_M})$ with $0 \leq n_c \leq h_c$ for all $c \in \mathcal{K}$ and $\sum_{c \in \mathcal{K}} n_c = N$. The stationary probability of observing the aggregated state with n_c elements of class c for all $c \in \mathcal{K}$ is:*

$$\pi_A(\mathbf{n}) = \frac{1}{G} \binom{N}{n_{c_1}, n_{c_2}, \ldots, n_{c_M}} \prod_{c \in \mathcal{K}} \prod_{j=0}^{n_c - 1} \lambda_c(j)\,,$$

where $\mathbf{n}$ belongs to the set of aggregated states

$$\mathcal{S}_{\mathcal{K},N} = \{\mathbf{n} : \sum_{c \in \mathcal{K}} n_c = N \textit{ and } 0 \leq n_c \leq h_c \ \forall c \in \mathcal{K}\}\,.$$

In what follows, we denote the normalising constant and the aggregated stationary distribution of the system model consisting of a set of traffic classes $\mathcal{K}$ and a window size N as $G_{\mathcal{K},N}$ and $\pi_{\mathcal{K},N}$, respectively. The marginal equilibrium distribution for each class is given by Lemma 3.5 and is expressed in terms of a ratio of the normalising constants of different models. Again the proof is given in the Appendix.

LEMMA 3.5. *The marginal stationary probability of observing exactly δ objects of class $d \in \mathcal{K}$ in the window, with $0 \leq \delta \leq h_d$, is:*

$$\pi^d_{\mathcal{K},N}(\delta) = \binom{N}{\delta} \left(\prod_{j=0}^{\delta-1} \lambda_d(j) \right) \frac{G_{\mathcal{K} \setminus \{d\}, N-\delta}}{G_{\mathcal{K},N}}\,,$$

where $G_{\mathcal{K} \setminus \{d\}, N-\delta}$ is the normalising constant associated with a model without class d and a window size of $N - \delta$.

3.3 Performance indices

In this section we introduce a set of performance indices and show how to compute them efficiently.

Definition 3.6 (*Admission rate*). The admission rate for a class $c \in \mathcal{K}$ is the rate associated with the event of transition from a state $\vec{x}$ with $|\vec{x}|_c = 0$ to a state $\vec{x}'$ with $|\vec{x}'|_c = 1$ when the model is in steady-state. The global admission rate is the sum of the admission rates for each $c \in \mathcal{K}$.

Definition 3.7 (*Rejection rate*). The rejection rate for a traffic class $c \in \mathcal{K}$ is the rate associated with the event of rejecting the arrival of class c because the number of objects of class c in the window is h_c. The global rejection rate is the sum of the rejection rates for each traffic class $c \in \mathcal{K}$.

The admission rate for a specific traffic class and the global admission rate can be computed as in Corollary 3.8, while the rejection rate for a specific traffic class and the global rejection rate can be computed as in Corollary 3.9. The proofs are given in the Appendix.

COROLLARY 3.8. *In steady-state, the admission rate for a traffic class $d \in \mathcal{K}$ is:*

$$X^d_{\mathcal{K},N} = \lambda_d(0) \frac{G_{\mathcal{K} \setminus \{d\}, N}}{G_{\mathcal{K},N}}\,, \tag{3}$$

and the global admission rate is:

$$X_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} \lambda_c(0)\frac{G_{\mathcal{K}\smallsetminus\{c\},N}}{G_{\mathcal{K},N}}\,, \tag{4}$$

COROLLARY 3.9. *In steady-state, the rejection rate for a traffic class $d \in \mathcal{K}$ is:*

$$Y^d_{\mathcal{K},N} = \lambda_d(h_d)\binom{N}{h_d}\prod_{j=0}^{h_d-1}\lambda_d(j)\frac{G_{\mathcal{K}\smallsetminus\{d\},N-h_d}}{G_{\mathcal{K},N}}\,, \tag{5}$$

and the global rejection rate is:

$$Y_{\mathcal{K},N} = \sum_{c\in\mathcal{K}}\lambda_c(h_c)\binom{N}{h_c}\prod_{j=0}^{h_c-1}\lambda_c(j)\frac{G_{\mathcal{K}\smallsetminus\{c\},N-h_c}}{G_{\mathcal{K},N}}\,. \tag{6}$$

By applying Lemma 3.5 we can compute the expected number of objects of a given traffic class in the window when the model is in steady-state and the throughput for each class.

COROLLARY 3.10. *In steady-state, the expected number of objects of class $d \in \mathcal{K}$ in the window is:*

$$\overline{N}^d_{\mathcal{K},N} = \sum_{\delta=1}^{h_d}\delta\binom{N}{\delta}\prod_{j=0}^{\delta-1}\frac{G_{\mathcal{K}\smallsetminus\{d\},N-\delta}}{G_{\mathcal{K},N}}\,. \tag{7}$$

COROLLARY 3.11. *In steady-state the throughput for a traffic class $d \in \mathcal{K}$ is:*

$$\lambda^*_d = \sum_{\delta=0}^{h_d-1}\lambda_d(\delta)\binom{N}{\delta}\left(\prod_{j=0}^{\delta-1}\lambda_d(j)\right)\frac{G_{\mathcal{K}\smallsetminus\{d\},N-\delta}}{G_{\mathcal{K},N}}\,. \tag{8}$$

Finally, we introduce an index to measure the fairness of the bandwidth allocation among the set of traffic classes.

Definition 3.12. Let $\mathcal{K}_1 \subseteq \mathcal{K}$ be a subset of the traffic classes whose elements have the same priority. Assume that the arrival rates for the classes in $\mathcal{K}_1$ are independent of the state of the window, i.e., for any $c \in \mathcal{K}_1$, $\lambda_c(j) = \lambda_c$ for all $0 \le j \le h_c$. The fairness index Φ_{cd} of a class c with respect to a class d with $c, d \in \mathcal{K}_1$ is defined as follows:

$$\Phi_{cd} = \min\left(\lambda_c - \lambda^*_c, \max(\lambda^*_d - \lambda^*_c, 0)\right).$$

The global fairness index for $\mathcal{K}_1$ is defined as:

$$\Phi_{\mathcal{K}_1} = \sum_{c\in\mathcal{K}_1}\frac{\lambda_c}{\lambda_{\mathcal{K}_1}}\sum_{d\in\mathcal{K}_1}\Phi_{cd}\,,$$

where $\lambda_{\mathcal{K}_1} = \sum_{c\in\mathcal{K}_1}\lambda_c$.

Intuitively, if $\Phi_{cd} > 0$ it means that $\lambda_c > \lambda^*_c$ and $\lambda^*_d > \lambda^*_c$, i.e., class c has been slowed down even if there is a class with the same priority which is consuming more bandwidth. In an ideal situation (i.e., that identified by the max-min fairness principle) this should not happen, and hence the value of Φ_{cd} should be as close to 0 as possible.

The next proposition states that when the fairness index is 0 we achieve the *max-min* fairness, i.e., a flow with low requirements is never slowed down while there exists another flow with the same priority which is using more resources [Bertsekas and Gallager 1992]. The proof is given in the Appendix.

PROPOSITION 3.13. *Let $\mathcal{K}_1 \subseteq \mathcal{K}$. The fairness index $\Phi_{\mathcal{K}_1}$ is 0 if and only if the allocation of the bandwidth $\lambda^*_{\mathcal{K}_1} = \sum_{c \in \mathcal{K}_1} \lambda^*_c$ is max-min fair.*

3.4 Computation of the normalising constant

The expression for the normalising constant given by Theorem 3.1 is computationally expensive and prone to numerical instability problems. In this section we provide an efficient algorithm for computing the normalising constant based on its convolution property. We define $\tau_{\mathcal{K}} \in \mathbb{N}$ as the maximum number of slots that the traffic classes in $\mathcal{K}$ would occupy in an infinite size window, i.e.,

$$\tau_{\mathcal{K}} = \sum_{c \in \mathcal{K}} h_c \, .$$

Notice that, given a partition $\mathcal{K}_1$ and $\mathcal{K}_2$ of the set of traffic classes $\mathcal{K}$, it clearly holds that $\tau_{\mathcal{K}} = \tau_{\mathcal{K}_1} + \tau_{\mathcal{K}_2}$. The proof of the next Lemma is given in the appendix.

LEMMA 3.14. *Let $\mathcal{K}$ be the set of traffic classes and let $\mathcal{K}_1$ and $\mathcal{K}_2$ be a partition of $\mathcal{K}$. Then, the normalising constant can be defined by the following recursive relation:*

$$G_{\mathcal{K},N} = \sum_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \binom{N}{j} G_{\mathcal{K}_1,N-j} G_{\mathcal{K}_2,j} \, . \tag{9}$$

Let us order the traffic classes $c_1, \ldots, c_M \in \mathcal{K}$ and let $\vec{h} = (h_{c_1}, \ldots, h_{c_M})$. We compute the normalising constant as shown in Algorithm 1 where we use the convention that array positions start from 1 and empty products have value 1. Notice that if $\mathcal{K}$ is a singleton, then the normalising constant $G_{\mathcal{K},\min(N,h_c)}$ can be computed easily as:

$$G_{\{c\},min(N,h_c)} = \prod_{j=0}^{\min(N,h_c)-1} \lambda_c(j) \, . \tag{10}$$

It is easy to see that the asymptotic complexity of the algorithm is $\mathcal{O}(MN^2)$ where M is the number of traffic classes and N is the window size.

4. PERFORMANCE EVALUATION IN MESH NETWORKS

In this section we use a Matlab implementation of Algorithm 1 to derive the performance indices of a mesh WSN where the data harvesting processes of the sensors are modelled by independent Poisson processes. The WSN consists of K motes that transmit data labelled with a class among the 20 available. By the superposition property of the Poisson processes, the mesh scenario allows us to assume that λ_c is the sum of all the rates of the sensors for class c traffic. We should stress on the fact that we are not assuming that all the sensors have the same harvesting rate for each class, but we observe that since each mote receives the packets generated by all the other motes of the WSN, then we can think of a global harvesting rate for each class given by the sum of those of the single sensors.

We consider two randomly generated scenarios: in the first case (S1) the total packet generation rate is 63.47 packets per unit of time with a standard deviation of 4.41, while in the second (S2) we have a total generation rate of 58.45 with a standard deviation of 1.35. The rates of the harvesting processes for each traffic class in S1 and S2 are shown in Table I. Moreover, we assume $\lambda_c(n_c) = \lambda_c$ for all classes $c \in \mathcal{K}$ and $0 \leq n_c < h_c$.

ALGORITHM 1: Convolution algorithm

input : $\mathcal{K}, \vec{h}, N$
output: $G_{\mathcal{K},N}$

$prevcol \leftarrow [0, \ldots, N]$;
$newcol \leftarrow [0, \ldots, N]$;
{*Initialise the first column*};
for $i \leftarrow 0$ **to** $\min(N, \tau_{\{c_1\}})$ **do**
  $newcol(i+1) \leftarrow \prod_{j=0}^{i-1} \lambda_{c_1}(j)$;
end
for $d \leftarrow 2$ **to** M **do**
  $prevcol \leftarrow newcol$;
  $newcol(1) \leftarrow 1$;
  $\mathcal{K}_1 \leftarrow \{c_1, \ldots, c_d\}$;
  {*Compute* $G_{\mathcal{K}_1,i}$ *for* $i = 1, \ldots \min(N, \tau_{\mathcal{K}_1})$ *and store the result in* $newcol(i+1)$};
  for $i \leftarrow 1$ **to** $\min(N, \tau_{\mathcal{K}_1})$ **do**
    $newcol(i+1) \leftarrow 0$;
    for $j \leftarrow \max(0, i - \tau_{\mathcal{K}_1 \setminus \{c_d\}})$ **to** $\min(N, \tau_{\mathcal{K}_1})$ **do**
      $newcol(i+1) \leftarrow newcol(i+1)$
        $+\binom{i}{j} prevcol(i-j+1) \prod_{z=0}^{j-1} \lambda_d(z)$;
    end
  end
end
$G_{\mathcal{K},N} \leftarrow newcol(N+1)$;

Class c	1	2	3	4	5	6	7	8	9	10
λ_c $(S1)$	1.00	1.30	1.50	1.80	3.8	1.20	1.50	1.72	1.12	8.00
λ_c $(S2)$	1.20	2.30	1.50	2.00	3.80	2.40	2.20	3.30	2.62	3.00
Class c	11	12	13	14	15	16	17	18	19	20
λ_c $(S1)$	1.00	1.30	1.35	6.78	4.10	1.20	1.66	1.70	1.44	20.0
λ_c $(S2)$	3.21	2.25	4.35	5.00	4.10	1.64	1.66	2.70	2.44	6.78

Table I. : Arrival rates for scenarios S1 and S2. The total arrival rate for S1 is $\lambda = 63.47$ with standar deviation of 4.42, while for S2 the total arrival rate is $\lambda = 58.45$ the same with a standard deviation of 1.35.

4.1 Impact of window size on admitted flow

In this section we assume the same $h_c = 1, 2$ for all classes and we study how the admitted flow depends on the window size. We assume that the sum of the transmission rate of the sensors in the mesh WSN are those specified for the scenarios S1 and S2 (see Table I) (see Figures 1-7). The plots in Figure 1 and 5 show the total throughput with respect to a given window size, for S1 and S2, respectively. We can observe the monotonic decrease of the total throughput with respect to the window size and the impact of the parameter h_c. Figures 2 and 6 show the plots of the throughput of the fastest and slowest classes for the two scenarios with different window sizes. As expected, the throughput of the fastest classes decreases much more quickly than those that are less aggressive in accordance with the max-min fairness principles.

Figure 3 and 7 show the impact of the window size on the fairness index for $h_c = 1$ and $h_c = 2$. We notice that, although that perfect max-min fairness is achieved when we admit almost all the streams (window size 1) or when they are all blocked we can see that the fairness index is always below 1.4

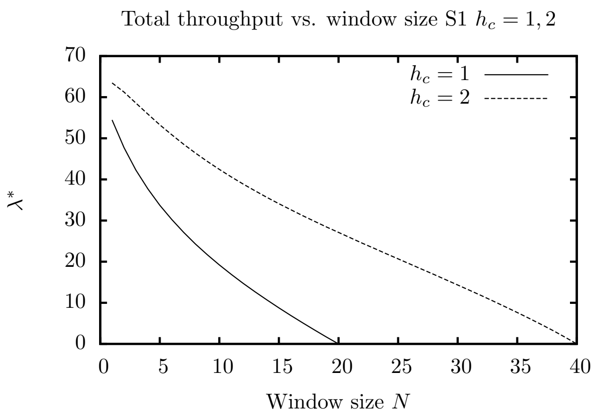

Fig. 1: Total admission rate for S1 and different values of h_c.

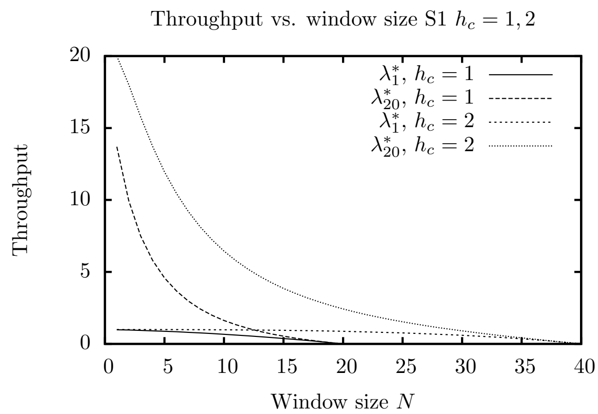

Fig. 2: Admission rate of the slowest and fastest class for S_1 and different values of h_c.

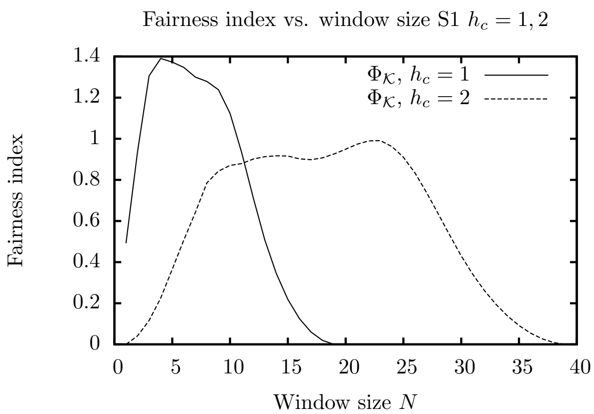

Fig. 3: Fairness index for different values of h_c as function of the window size.

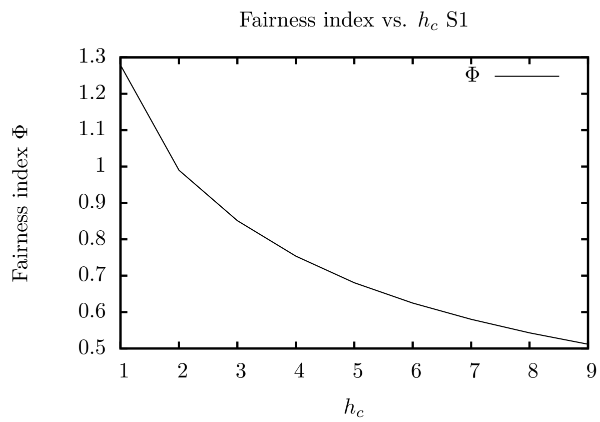

Fig. 4: Fairness index for different values of h_c and fixed window size.

in S1 and 2.5 in S2, therefore we can conclude that the effect of the admission control algorithm is satisfactory.

4.2 Impact of h_c on the fairness index

In this section we consider the scenario S1 and we assume $T = 25$. We configure the window size such that λ^* is maximum under the constraint $\lambda^* < T$. We study the system for $h_c = 1, \ldots, 9$ with $c \in \mathcal{K}$. The results are shown in Table II and the plot of $\Phi_{\mathcal{K}}$ as function of h_c is shown in Figure 4. We can see that the fairness is improved by larger values of h_c but, in order to control the maximum throughput, it requires larger windows and more memory. This may be in contrast with the low resource requirements of the motes. As a consequence, a trade-off between memory occupancy and maxi-min fairness arises.

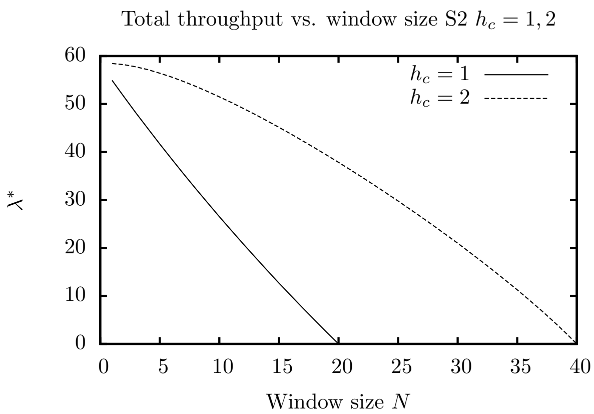

Fig. 5: Total admission rate for S2 and different values of h_c.

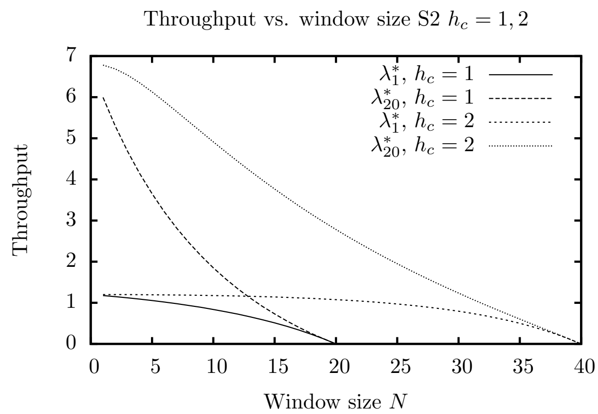

Fig. 6: Admission rate of the slowest and fastest class for S_2 and different values of h_c.

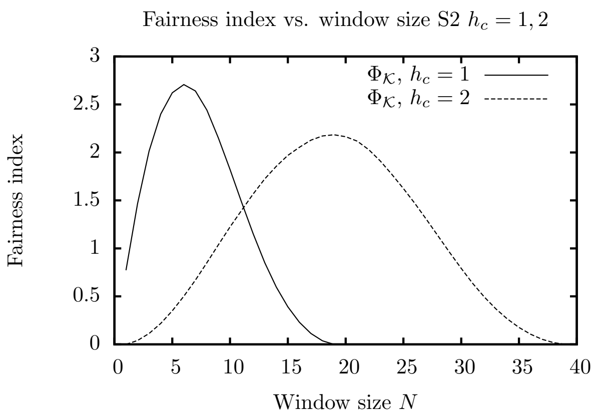

Fig. 7: Fairness index in S2 and different values of h_c.

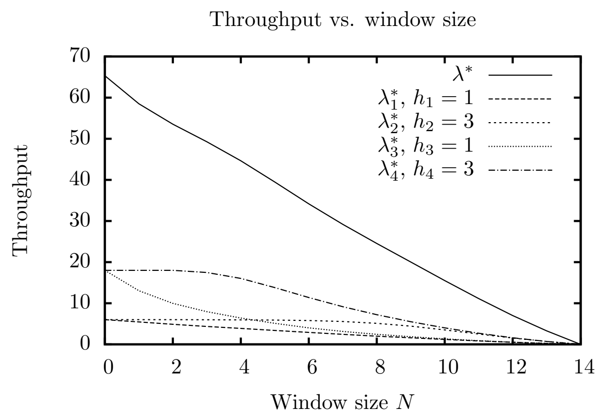

Fig. 8: Throughput of classes with same rate but different priority.

4.3 Different priority traffic streams

In this part we study a scenario consisting of 10 classes with the following packet generation rates:

$$(6.0, 6.0, 18.0, 18.0, 3.8, 1.2, 1.5, 1.72, 1.12, 8.0)\,.$$

We compare the bandwidth allocation to two pairs of streams: $1, 2$ and $3, 4$.

Streams 1 and 2 (resp. 3 and 4) have the same rates but the priority of 2 (resp. 4) is higher than that of 1 (resp. 3). We model this by setting $h_1 = h_3 = 1$ and $h_2 = h_4 = 3$. In Figure 8 we show the throughput of the four streams together with the total throughput. We notice that while with the increasing of the control window's size the total throughput obviously decreases, the reduction of the bandwidth assigned to the high priority traffic streams 2 and 4 is much slower than that experienced by the lower priority streams 1 and 3.

h_c	N	λ^*	Φ
1	8	24.2832	1.2786
2	22	24.5006	0.9897
3	38	24.1746	0.8513
4	54	24.5124	0.7537
5	70	24.9674	0.6807
6	87	24.9928	0.6249
7	105	24.7372	0.5800
8	123	24.6059	0.5430
9	140	24.8422	0.5117

Table II. : Impact of h_c on the fairness index

5. SIMULATIONS

The results shown in the previous sections are based on the assumption that the WSN is a mesh network, i.e., every node is in the transmission range of every other node. However, in practice, this is hard to achieve especially in large scale WSNs. In general, routing protocols are design in such a way that the nodes are not exposed to the same traffic intensity. For instance, the nodes near the network sinks tend to be more stressed than others, or in case of tree-structured networks the traffic may be unbalanced due to the hierarchical structure of the routing protocol. In this section we study the sensitivity of the analytical results previously proposed under the hypothesis of dealing with a mesh network. We resort to a stochastic simulation to show that in a connected ad-hoc network the average performance indices are not very sensitive to this assumption and hence the model proposed in Section 3 can be used to estimate the system throughput and fairness quite accurately. To this aim, we developed an ad-hoc simulator that abstracts out the implementation details of the routing protocols, but assumes that the traffic is unbalanced, i.e., some nodes will handle more packets than others in the long run. Moreover, we set a maximum transmission radius so that the mesh assumption does not hold.

All the simulations, unless differently specified, consist of 30 independent experiments whose warm-up phase has been removed according to the Welch's procedure [Welch 1981]. The confidence intervals have a confidence level of 98%.

5.1 Validation of the simulator

The first step consists in validating the simulator. To this aim, we simulate a mesh network that satisfies all the assumptions of the model proposed in Section 3. Table III shows that the comparison of the results given by the analytical model and the estimates of the simulation. We observe that the analytical values always fall in the confidence intervals whose relative error is very small. Thus, we consider the simulator validated.

5.2 Simulation of general (non-mesh) networks

We now consider a WSN consisting of 300 sensors deployed in an area of $1000\text{m}\times1000\text{m}$ whose transmission range is 100m. The location of the sensors is random with uniform distribution and we assume that the network is connected, i.e., there is at least one single- or multi-hop route connecting each pair of motes of the net. In this experiment, the transmission rates of the motes for each class of data are sampled from exponential distributions whose means are the values given in the description of Scenario 1 (see Table I). The throughput and the fairness index are shown in Tables IV. We observe that the analytical model provides a high level of accuracy for the estimation of the performance indices in this case, with a relative error lower than 2%, even if the packet generation rates for the same class are

h_c	N	λ^*	λ^s	λ^s_{lower}	λ^s_{upper}	Φ^*	Φ^s
1	8	24.2832	24.2814	24.2618	24.3011	1.2786	1.2788
2	22	24.5006	24.5021	24.4861	24.5181	0.9897	0.9896
3	38	24.1746	24.1746	24.1534	24.1958	0.8513	0.8513
4	54	24.5124	24.5141	24.4959	24.5323	0.7537	0.7536
5	70	24.9674	24.9649	24.9463	24.9835	0.6807	0.6808
6	87	24.9928	24.9909	24.9748	25.0071	0.6249	0.6250
7	105	24.7372	24.7368	24.7173	24.7563	0.5800	0.5800
8	123	24.6059	24.6069	24.5895	24.6242	0.5430	0.5430
9	140	24.8422	24.8413	24.8214	24.8612	0.5117	0.5118

Table III. : Validation of the simulator for a mesh network. N identical nodes are deployed and all the classes have the same h_c. The packet generation rate per class are shown in Table I (Scenario 1), the throughput and the fairness index obtained by the model are λ^* and Φ^*, respectively, and those estimated by the simulation are λ^s and Φ^s, respectively.

different for the various motes. We conclude that if the WSN is sufficiently dense to be connected the total throughput of the proposed admission control algorithm depends mainly on the expected average harvesting rates of the nodes in the network and is not very sensitive to their distribution.

h_c	N	λ^*	λ^s	λ^s_{lower}	λ^s_{upper}	Φ^*	Φ^s
1	8	24.2832	24.1775	24.1587	24.1962	1.2786	1.3114
2	22	24.5006	24.2029	24.1822	24.2235	0.9897	1.0178
3	38	24.1746	23.9317	23.9085	23.9549	0.8513	0.8521
4	54	24.5124	24.5962	24.5727	24.6197	0.7537	0.7525
5	70	24.9674	25.0125	24.9970	25.0281	0.6807	0.6897
6	87	24.9928	24.8864	24.8680	24.9047	0.6249	0.6046
7	105	24.7372	24.7002	24.6768	24.7235	0.5800	0.5938
8	123	24.6059	24.6298	24.6109	24.6488	0.5430	0.5509
9	140	24.8422	25.0960	25.0738	25.1183	0.5117	0.5261

Table IV. : Simulated results with confidence interval at 98% confidence level.

Figures 9 and 10 (11 and 12) show the comparison of the simulation estimates with the analytical results for S1 (S2). The confidence intervals are too small to be displayed in the figures. We can see that, despite the stricter hypotheses required by the analytical model are violated, the results that we can obtain by its analysis are very accurate.

Figures 13-16 show the throughput for the slowest and fastest classes of S1 for $h_c = 1$ and $h_c = 2$ for all c. We observe that the simulation estimates confirm the analytical indices in showing that the throughput of the greediest class (the fastest) is dropped much more quickly than that of the slowest class. This corresponds to the principle that in case of two streams with the same priority, we desire to contain the needs of the fastest one before reducing the resources used by the slowest one. The throughput of S2 for the fastest and slowest classes is studied in Figures 17-20. Recall that, with respect to S1, the standard deviation of the distribution of the harvesting rates among the classes is lower. As a consequence, the throughput of class 20 is reduced slower than in S1. Finally, we study the total throughput of the network (consisting of all the classes) for the two scenarios and $h_c = 1, 2$ in Figures 21-24. As we can see, the simulation estimates are almost indistinguishable from the analytical results despite the relaxation of the assumptions done in the simulation model.

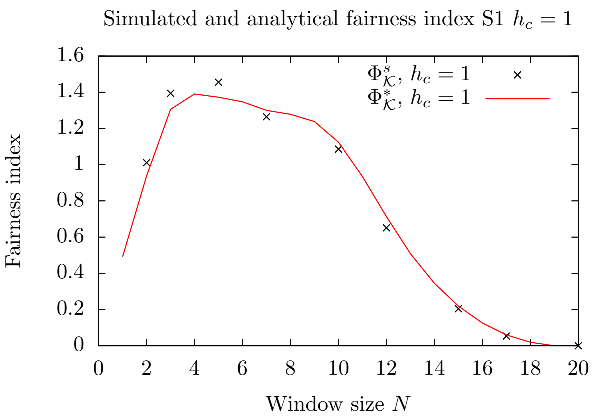

Fig. 9: Fairness index in Scenario 1 with $h_c = 1$.

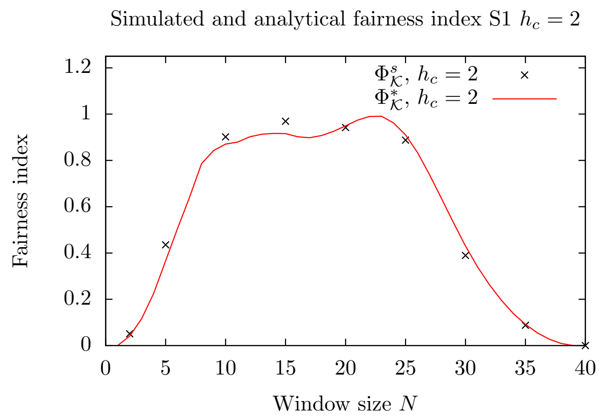

Fig. 10: Fairness index in Scenario 1 with $h_c = 2$.

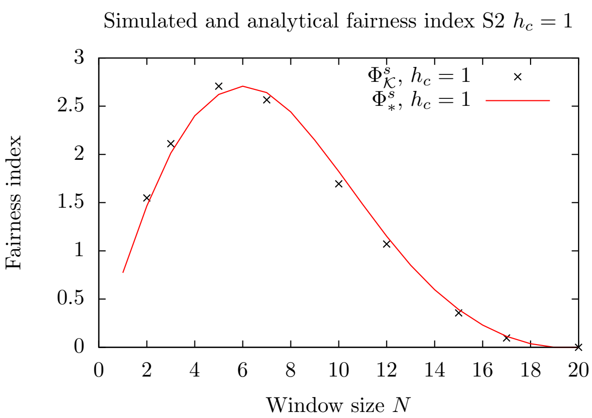

Fig. 11: Fairness index in Scenario 2 with $h_c = 1$.

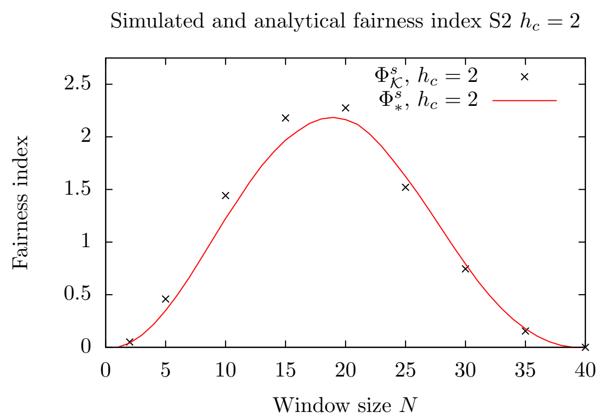

Fig. 12: Fairness index in Scenario 2 with $h_c = 2$.

6. RELATED WORK

We discuss the works related to our contribution in two steps. First, we compare our approach with other works which address the problem of congestion control in WSNs (with or without priorities). Secondly, we compare our theoretical contribution in terms of CTMC analysis with respect to the literature.

The problem of bandwidth assignment in wireless sensor networks have been addressed by a large number of papers (see [Chitnis et al. 2009] and the references therein). In [Woo and Culler 2001] the authors introduce the Adaptive Rate Control (ARC)scheme. In this scheme each mote estimates the number of downstreams and the bandwidth is split proportionally among the local and router through traffic. In [Tan et al. 2015] the authors solve an optimisation problem in order to assign the resources in a networks where soft and hard constraints are simultaneously required. To this aim they introduce an effective algorithm whose optimality is proved under mild conditions. With respect to this work, the approach that we propose is totally decentralized and quickly reactive to changes in the traffic needs or in the network topology. The solutions proposed in [Sankarasubramabiam et al. 2003; Wan et al. 2003] use control packets to avoid congestion. Another important contribution is the rate control scheme in-

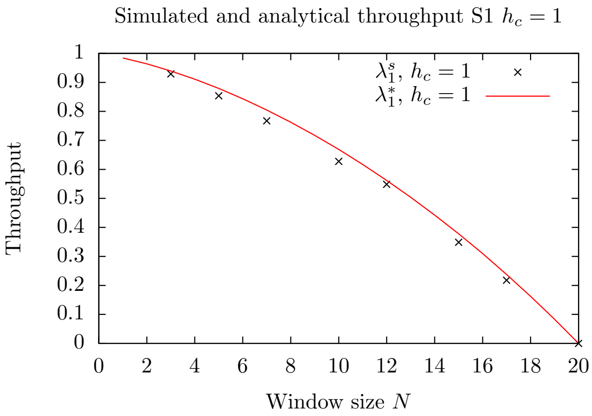

Fig. 13: Total throughput of the slowest class in Scenario 1 with $h_c = 1$.

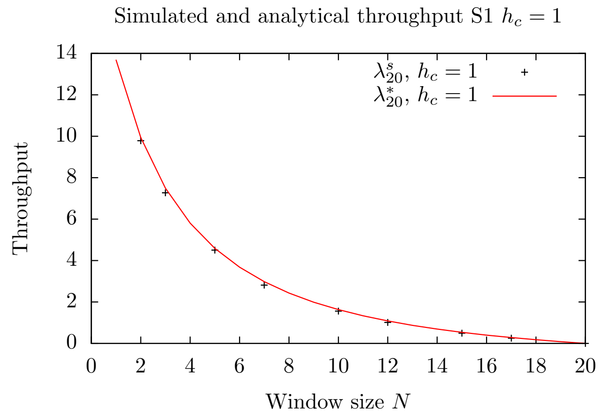

Fig. 14: Total throughput of the fastest class in Scenario 1 with $h_c = 1$.

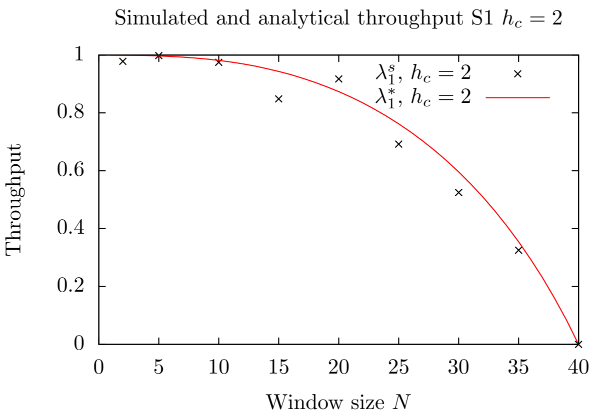

Fig. 15: Total throughput of the slowest class in Scenario 1 with $h_c = 2$.

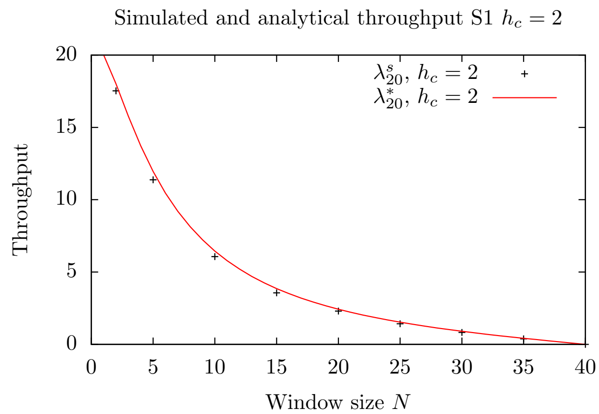

Fig. 16: Total throughput of the fastest class in Scenario 1 with $h_c = 2$.

troduced in [Ee and Bajcsy 2004] where the authors consider a tree-structured network and each mote estimates the average rate at which packets can be sent and divide it by the number of children motes downstream obtaining the maximum flow associated with each child mote. In [He et al. 2003] the authors present the SPEED protocol to achieve soft real time communication in WSNs. SPEED exploits little knowledge about the network and provides a mechanism for the packet routing that allows for a fair delivery time for the data packets. All these congestion control schemes do not consider traffic priorities, and require the transmission of control packets (which may be done by piggybacking). In [Khan et al. 2012] the authors propose a probabilistic approach to control the bandwidth assignment in WSNs based on the IEEE 802.15.4 standard in a tree-structured WSN. The protocol aims at obtaining the fairness among the nodes rather than among the traffic types. Traffic priorities are considered in [Caccamo et al. 2002; Cherian and Nair 2014] that exploit earliest deadline first scheduling and priority queues, respectively. The protocols proposed in [Caccamo et al. 2002] require an accurate knowledge of the network topology and divide the motes into cells. Then, the communication intra-cell and extra-cell are handled in different ways. The scheme proposed in [Cherian and Nair 2014] simply proposes a priority scheduler for each mote. Although this allows a single node to use priority for its own transmission, the correlation among the motes is not taken into account.

One major problem in the resource allocation in WSNs consists deciding which nodes should take care of forwarding the packets in a multi-hop routing protocol. Although energy preservation is outside the main scope of this paper, it is worth of notice that many works are devoted to the analysis of this problem (see, e.g., [Tan and Wu 2016] and the references therein). In this paper we have analysed the effects of FACW with respect to the bandwidth allocation problem, however we point out that since nodes with larger windows tend to transmit less packets than those with small windows, similarly to the solutions proposed in [Tan and Wu 2016], FACW can be effectively used in order to reduce the energy consumption in stressed nodes that, consequently, will focus only on the traffic associated with higher priority classes.

Analytical models of WSNs have been widely studied in the literature, with different aims (see, e.g., [Degirmenci et al. 2013; Tschaikowski and Tribastone 2017]). However, from the point of view of the stochastic analysis, the most related works are those presented in [King 1971; Pan et al. 2000]. In [Pan et al. 2000] the authors introduce CHOKe, i.e., a congestion control mechanism that allows the approximate fair sharing of a bandwidth among a set of competing customers. The authors propose a model for the analysis of *back-CHOKe* which is based on maintaining a window with the latest N packets arrived at the bottleneck. The packets coming from a source which is present in the window are discarded. Packet arrivals occur according to independent homogeneous Poisson processes, i.e., the model is a continuous time version of King's model for the FIFO cache under the Independence Reference Model assumption (IRM) [King 1971]. With respect to these papers, we propose a more sophisticated model in which the window may contain a number of replicas that depends on the traffic type. Moreover, we give an efficient algorithm to compute the performance measures that implements a convolution on the finite state space of the CTMC. Indeed, the algorithm developed in [Fagin and Price 1978] is not applicable to our model due to the possible presence of duplicated items in the window. Finally, with respect to the models studied in [King 1971; Fagin and Price 1978; Pan et al. 2000], we relax the requirements of the IRM by allowing the rate of the Poisson processes generating the data at the motes to depend on the window state.

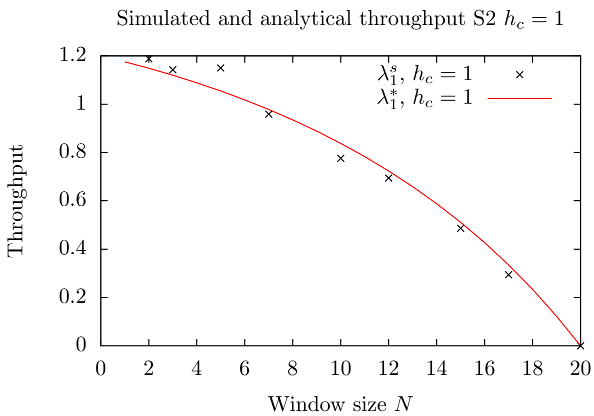

Fig. 17: Total throughput of the slowest class in Scenario 2 with $h_c = 1$.

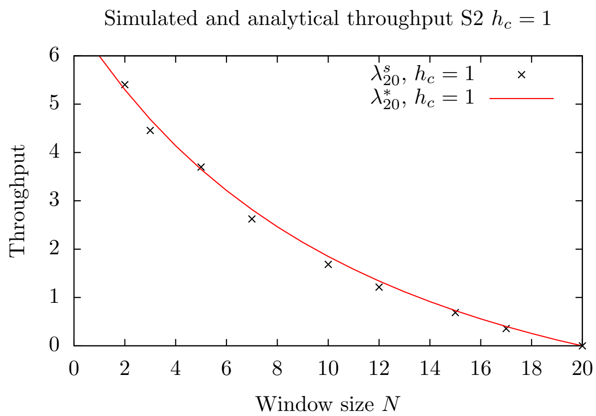

Fig. 18: Total throughput of the fastest class in Scenario 2 with $h_c = 1$.

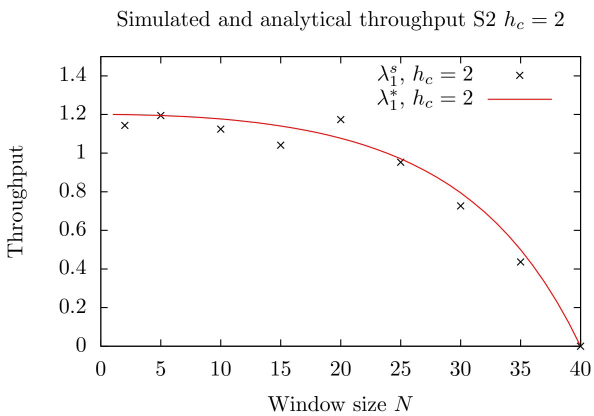

Fig. 19: Total throughput of the slowest class in Scenario 2 with $h_c = 2$.

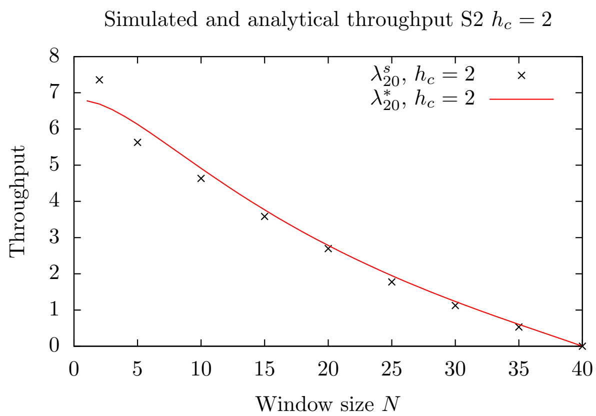

Fig. 20: Total throughput of the fastest class in Scenario 2 with $h_c = 2$.

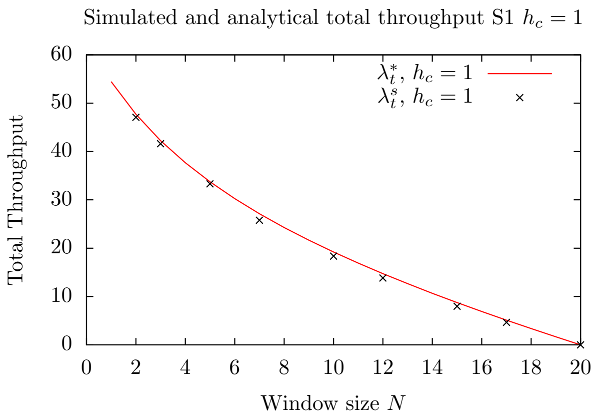

Fig. 21: Total throughput in Scenario 1 with $h_c = 1$.

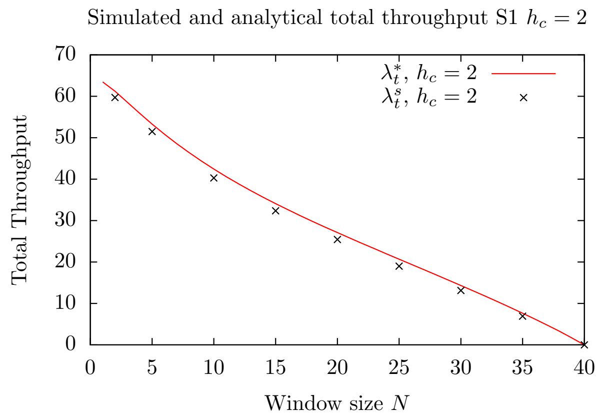

Fig. 22: Total throughput in Scenario 1 with $h_c = 2$.

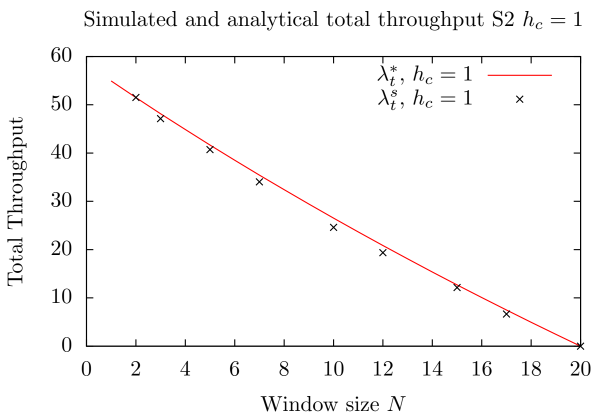

Fig. 23: Total throughput in Scenario 2 with $h_c = 1$.

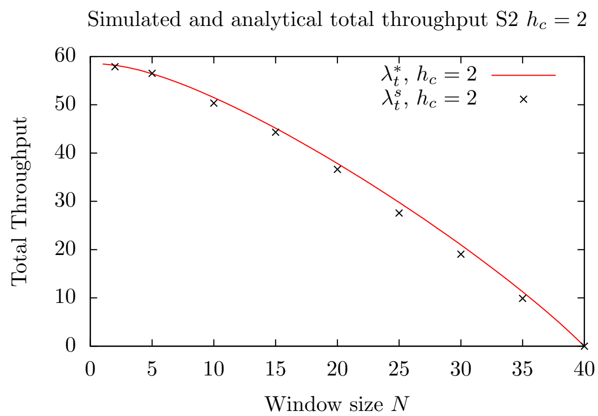

Fig. 24: Total throughput in Scenario 2 with $h_c = 2$.

7. CONCLUSION

In this paper we have proposed an allocation control scheme, named FACW, for the bandwidth assignment in WSNs. FACW is easy to implement in an actual protocol, consumes few resources in the motes and does not require extra control traffic in the WSN. We showed that it is able to handle traffic streams with different priorities and can reach a good level of fairness among streams with the same priority. Its main idea consists in maintaining at each mote a window with the latest traffic types perceived and dropping packets of the types that have reached their maximum population in the window. Under the assumption of Poisson generated traffic, we have proposed a model which is analytically tractable and gave an algorithm to efficiently derive the performance indices. The model belongs to the class of product-form models and hence can be used to tackle the problems of the state space explosion which is typical of CAS [Feng et al. 2016]. By an extensive set of simulations, we have shown that the model is robust with respect to the assumption that the WSN forms a mesh network, and hence can be applied to study large wireless networks.

REFERENCES

AKYILDIZ, I. F. AND KASIMOGLU, I. H. 2004. Wireless sensor and actor networks: research challenges. *Ad Hoc Networks 2,* 4, 351 – 367.

BERTSEKAS, D. AND GALLAGER, R. 1992. *Data networks*. Prentice Hall.

CACCAMO, M., ZHANG, L., LUI, S., AND BUTTAZZO, G. 2002. An implicit prioritized access protocol for wireless sensor networks. In *Proc. of 23rd IEEE Real-Time Systems Symposium (RTSS)*. 39–48.

CHERIAN, M. AND NAIR, T. R. G. 2014. Priority based bandwidth allocation in wireless sensor networks. *Int. J. of Computer Networks & Communications (IJCNC) 6,* 6, 119–128.

CHITNIS, M., PAGANO, P., LIPARI, G., AND LIANG, Y. 2009. A survey on bandwidth resource allocation and scheduling in wireless sensor networks. In *Proc. of IEEE Int. Conf. on Network-Based Information Systems*. 121–128.

DEGIRMENCI, G., KHAROUFEH, J. P., AND BALDWIN, R. O. 2013. On the performance evaluation of query-based wireless sensor networks. *Perform. Eval. 70,* 2, 124–147.

EE, C. T. AND BAJCSY, R. 2004. Congestion control and fairness for many-to-one routing in sensor networks. In *Proc. of the 2Nd Int. Conf. on Embedded Networked Sensor Systems*. 148–161.

FAGIN, R. AND PRICE, T. G. 1978. Efficient calculation of expected miss ratios in the independent reference model. *SIAM J. Comput. 7,* 3, 288–297.

FENG, C., HILLSTON, J., AND GALPIN, V. 2016. Automatic moment-closure approximation of spatially distributed collective adaptive systems. *ACM Trans. Model. Comput. Simul. 26,* 4, 26:1–26:22.

FRESCHA, A. 2015. Collective adaptive systems. In *Proc. of UBICOMP/ISWC '15 Adjunct*. 893–895.

GELENBE, E. AND MARIN, A. 2015. Interconnected wireless sensors with energy harvesting. In *Proc. of Int. Conf. on Analytical Stochastic Modelling Techniques and Applications (ASMTA)*. 87–99.

HAHNE, E. L. 1991. Round-robin scheduling for max-min fairness in data networks. *IEEE Journal on Selected Areas in Communications 9*, 1024–1039.

HANZALEK, Z. AND JURĈIK, P. 2010. Energy efficient scheduling for cluster-tree wireless sensor networks with time-bounded data flows: Application to ieee 802.15.4/zigbee. *Industrial Informatics, IEEE Transactions on 6,* 3, 438 – 450.

HE, T., STANKOVIC, J. A., LU, C., AND ABDELZAHER, T. 2003. SPEED: A stateless protocol for real-time communication in sensor networks. In *Proc. 23rd IEEE Int. Conf. on Distributed Computing Systems*. 46–55.

KELLY, F. 1979. *Reversibility and stochastic networks*. Wiley, New York.

KHAN, D., NEFZI, B., SANTINELLI, L., AND SONG, Y. 2012. Probabilistic bandwidth assignment in wireless sensor networks. In *Wireless Algorithms, Systems, and Applications*. Lecture Notes in Computer Science Series, vol. 7405. Springer Berlin / Heidelberg, 631–647.

KING, W. F. 1971. Analysis of paging algorithms. In *Proc. of IFIP Congr.*

LOPES PEREIRA, R., TRINIDADE, J., GONCALVES, F., SURESH, L., BARBOSA, D., AND VAZAO, T. 2014. A wireless sensor network for monitoring volcano-seismic signals. *Nat. Hazards Earth Syst. Sci. 14*, 3123–3142.

MAINWARING, A., CULLER, D., POLASTRE, J., SZEWCZYK, R., AND ANDERSON, J. 2002. Wireless sensor networks for habitat monitoring. In *Proc. of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*. ACM, 88–97.

MARIN, A. AND ROSSI, S. 2014a. On discrete time reversibility modulo state renaming and its applications. In *Proc. of Valuetools 2014*.

MARIN, A. AND ROSSI, S. 2014b. On the relations between lumpability and reversibility. In *Proc. of the IEEE 22nd Int. Symp. on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'14)*. 427–432.

MARIN, A. AND ROSSI, S. 2016a. On the relations between Markov chain lumpability and reversibility. *Acta Informatica*. Available online.

MARIN, A. AND ROSSI, S. 2016b. Priority-based bandwidth allocation in wireless sensor networks. In *Proceedings of the 9th EAI International Conference on Performance Evaluation Methodologies and Tools*. VALUETOOLS'15. 137–144.

PAN, R., PRABHAKAR, B., AND PSOUNIS, K. 2000. CHOKe, A stateless active queue management scheme for approximating fair bandwidth allocation. In *Proc. of IEEE INFOCOM '00*. IEEE Computer Society Press, Washington, DC, 942–951.

SANKARASUBRAMABIAM, Y., AKAN, O., AND AKYILDIZ, I. 2003. Event-to-sink reliable transport in wireless sensor networks. In *Proc. of the 4th ACM Symposium on Mobile Ad Hoc Networking & Computing, MobiHoc*. 177–188.

TAN, L. AND TANG, S. 2017. Energy harvesting wireless sensor node with temporal death: novel models and analyses. *IEEE/ACM Transactions on Networking 25,* 2, 896–909.

TAN, L. AND WU, M. 2016. Data reduction in wireless sensor networks: a hierarchical LMS prediction approach. *IEEE Sensors Journal 16,* 6, 1708–1715.

TAN, L., ZHU, Z., GE, F., AND XIONG, N. 2015. Utility maximization resource allocation in wireless networks: Methods and algorithms. *IEEE Trans. on Systems, Man and Cybernetics: Systems 45,* 7, 101–1034.

TANG, S. AND TAN, L. 2017. Reward rate maximization and optimal transmission policy of EH device with temporal death in EH-WSNs. *IEEE Trans. on Wireless Communications 16,* 2, 1157–1167.

TSCHAIKOWSKI, M. AND TRIBASTONE, M. 2017. Spatial fluid limits for stochastic mobile networks. *Perform. Eval. 109*, 52–76.

TUNC, C. AND AKAR, N. 2017. Markov fluid queue model of an energy harvesting IoT device with adaptive sensing. *Perform. Eval. 111*, 1–16.

WAN, C., EISENMAN, S. B., AND CAMPBELL, A. T. 2003. Coda: Congestion detection and avoidance in sensor networks. In *First ACM Conf. on Embedded Networked Sensor Systems*. 266–279.

WELCH, P. D. 1981. On the problem of the initial transient in steady-state simulations. Tech. rep., IBM Watson Research Center, Yorktown Heights, NY.

WHITTLE, P. 1986. *Systems in stochastic equilibrium*. John Wiley & Sons Ltd.

WOO, A. AND CULLER, D. E. 2001. A transmission control scheme for media access in sensor networks. In *Proc. of Seventh Int. Conf. on Mobile Computing and Networking*. 221–235.

YI, W. Y., LO, K. M., MAK, T. LEUNG, K. S., LEUNG, Y., , AND MENG, M. L. 2015. A survey of wireless sensor network based air pollution monitoring systems. *Sensors (Basel Zwitzerland) 15*, 31392–31427.

8. APPENDIX

PROOF OF THEOREM 3.1. The proof is structured as follows: we first make a claim that $X(t)$ is ϱ-reversible and then, from Proposition 3.3, we derive Expression (1) of the stationary distribution. Finally, by using Proposition 3.2 we prove the claim.

CLAIM 1. *The process $X(t)$ for the FIFO policy is ϱ-reversible with respect to the renaming ϱ on S defined by $\varrho(\vec{x}) = \vec{x}^R$ where $\vec{x} = (x_1, \dots, x_N)$ and $\vec{x}^R = (x_N, \dots, x_1)$.*

We assume that $h_c = N$ for all the traffic classes $c \in \mathcal{K}$. We will see later that this assumption does not limit the validity of this proof. Assuming Claim 1 we use Proposition 3.3 to derive the expression of the stationary distribution π. Let us take a reference state $\vec{x}_0 = \vec{c}_1^{\,N}$ the N-sized vector whose entries are all equal to c_1, and let us derive the stationary probability of a general state $\vec{x} \in \mathcal{S}$. Consider the sequence of arrivals that starting from state $\vec{x}$ take the model to state $\vec{x}_0$ consisting in the arrival of exactly N objects of class c_1. We denote this path as follows:

$$\vec{x}_1 \equiv \vec{x} \xrightarrow{c_1} \vec{x}_2 \xrightarrow{c_1} \vec{x}_3 \cdots \xrightarrow{c_1} \vec{x}_{N+1} \equiv \vec{x}_0\,,$$

where we have labelled the arrows with the arriving classes. Notice that the reversed path from $\vec{x}_0^R = \vec{c}_1^{\,N}$ to $\vec{x}^R = (x_N, \dots, x_1)$ exists in the same process and is formed by the arrival of the sequence of traffic classes $x_1, x_2, \dots x_N$. Suppose that $\vec{x}$ has $K \le N$ objects of class c_1 in positions $i_1 < i_2 < .. < i_K \le N$.

The product of the rates in the forward path must take into account that the number of c_1 in the window starts from K and keeps increasing a unity at each arrival with the exception of the case in which an object of class c_1 is discarded. The c_1 in position i_k will be in position N after $N - i_k$ arrivals. The arrival $N - i_k + 1$ will leave the same number of class c_1 objects in the queue which is $N - i_k + 1$ due to the arrival plus the $k - 1$ which are with index lower than i_k. Therefore, the product of the rates of the forward path is:

$$\prod_{j=K}^{N-1} \lambda_{c_1}(j) \cdot \prod_{k=1}^{K} \lambda_{c_1}(N - i_k + k)\,. \tag{11}$$

The product of the rates in the reversed path is:

$$\left(\prod_{c \in \mathcal{K} \setminus \{c_1\}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j) \right) \cdot \prod_{k=1}^{K} \lambda_{c_1}(N - i_k + k)\,, \tag{12}$$

where the first factor is due to the arrivals of class $c \neq c_1$ objects while the second is due to class c_1 object arrival. Indeed if in $\vec{x}$ the objects of class c_1 are present in position i_k, $1 \leq k \leq K$, this means that the i_k-th arrival will be a c_1. The number of occurrences of c_1 in the window is $N - i_k + 1$ (due to the previus $i_k - 1$ arrivals) plus $k - 1$ due to the c_1 objects already arrived. Using Proposition 3.3, by Equations (11) and (12), we can derive $\pi(\vec{x})$:

$$\pi(\vec{x}) = \pi(\vec{x}_0) \frac{\prod_{c \in \mathcal{K} \setminus \{c_1\}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)}{\prod_{j=K}^{N-1} \lambda_{c_1}(j)}\,. \tag{13}$$

This says that if Claim 1 is true, then Equation (1) is the stationary distribution of $X(t)$. Indeed, by Equation (1):

$$\pi(\vec{x}_0) = \frac{1}{G} \prod_{j=0}^{N-1} \lambda_{c_1}(j)$$

and then by Equation (13) we can write

$$\pi(\vec{x}) = \frac{1}{G} \frac{\prod_{j=0}^{N-1} \lambda_{c_1}(j) \prod_{c \in \mathcal{K} \setminus \{c_1\}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)}{\prod_{j=K}^{N-1} \lambda_{c_1}(j)}$$

that is equal to

$$\pi(\vec{x}) = \frac{1}{G} \prod_{j=0}^{K-1} \lambda_{c_1}(j) \prod_{c \in \mathcal{K} \setminus \{c_1\}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)\,.$$

Now since $|\vec{x}|_{c_1} - 1 = K$, we obtain

$$\pi(\vec{x}) = \frac{1}{G} \prod_{c \in \mathcal{K}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)$$

proving that Equation (1) is indeed the stationary distribution of $X(t)$.

Let us now prove that Claim 1 is true by using Proposition 3.2. We show that Equation (1) satisfies the detailed balance equations for the following models:

(1) FIFO with $h_c = N$ for all $c \in \mathcal{K}$, which is the case we used to derive the candidate expression,

(2) FIFO with arbitrary $1 \leq h_c < N$ for some $c \in \mathcal{K}$.

Let $\vec{x} \in \mathcal{S}$ be $(x_1, \dots x_N)$. We distinguish two cases.
Case 1. The detailed balance equation becomes:

$$\pi(\vec{x})\lambda_d(|\vec{x}|_d) = \pi(d, x_1, \dots, x_{N-1})\lambda_{x_N}(|\vec{x}|_{x_N} - \delta_{x_N \neq d}) .$$

If $x_N \neq d$, by substituting the expression (1) of π we obtain:

$$\prod_{c \in \mathcal{K}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j)\lambda_d(|\vec{x}|_d) = \prod_{c \in \mathcal{K}} \prod_{j=0}^{|\vec{x}|_c - 1} \lambda_c(j) \frac{\lambda_d(|\vec{x}|_d)}{\lambda_{x_N}(|\vec{x}|_{x_N} - 1)} \lambda_{x_N}(|\vec{x}|_{x_N} - 1) ,$$

which is an identity. If $x_N = d$ the detailed balance equation is trivially an identity since $\pi(\vec{x}) = \pi(x_N, x_1, \dots, x_{N-1})$ and also the transition rates are identical.
Case 2. Let us consider now the case of $1 \leq h_c < N$ for some c. Observe that if there exists a transition from $\vec{x}$ to a different state due to the arrival of a class d, then $|\vec{x}|_d < h_d$. If $x_N = d$, this implies that also the reversed transition is possible and we already showed that the detailed balance equation is satisfied. If $x_N \neq d$, then clearly $|\vec{x}|_{x_N} \leq h_{x_N}$ that implies that the reversed transition is allowed since it occurs in a state with $|\vec{x}|_{x_N} - 1$ objects of class x_N. □

PROOF OF COROLLARY 3.4. The proof follows from the fact that all the window states with the same traffic class population (but different order) have the same stationary probability. Therefore, the stationary probability of an aggregated state is just the stationary probability of one window configuration multiplied by the number of configurations with the same traffic class population. □

PROOF OF LEMMA 3.5. By Corollary 3.4 we have:

$$\pi^d_{\mathcal{K},N}(\delta) = \sum_{\substack{\mathbf{n} \in \mathcal{S}_{\mathcal{K},N} \\ n_d = \delta}} \pi_{\mathcal{K},N}(\mathbf{n}) = \frac{1}{G_{\mathcal{K},N}} \sum_{\substack{\mathbf{n} \in \mathcal{S}_{\mathcal{K},N} \\ n_d = \delta}} \binom{N}{\mathbf{n}} \prod_{c \in \mathcal{K}} \prod_{j=0}^{n_c - 1} \lambda_c(j) ,$$

which can be conveniently rewritten as:

$$\frac{1}{G_{\mathcal{K},N}} \left(\prod_{j=0}^{\delta-1} \lambda_d(j) \right) \frac{(N-\delta+1)(N-\delta+2)\cdots N}{\delta!} \cdot \sum_{\substack{\mathbf{n} \in \mathcal{S}_{\mathcal{K},N} \\ n_d = \delta}} \frac{(N-\delta)!}{\prod_{c \in \mathcal{K} \setminus \{d\}} n_c!} \prod_{c \in \mathcal{K} \setminus \{d\}} \prod_{j=0}^{n_c - 1} \lambda_c(j) .$$

The result follows by observing that

$$\binom{N}{\delta} = \frac{(N-\delta+1)(N-\delta+2)\cdots N}{\delta!}$$

and that the summatory is the definition of the normalising constant for a model with traffic classes $\mathcal{K} \setminus \{d\}$ and window size $N - \delta$. □

PROOF OF COROLLARY 3.8. First observe that $X^d_{\mathcal{K},N} = \lambda_d(0)\pi^d_{\mathcal{K},N}(0)$ and $X_{\mathcal{K},N} = \sum_{c \in \mathcal{K}} X^c_{\mathcal{K},N}$. By Lemma 3.5, we obtain:

$$X^d_{\mathcal{K},N} = \lambda_d(0) \binom{N}{0} \left(\prod_{j=0}^{-1} \lambda_d(j) \right) \frac{G_{\mathcal{K} \setminus \{d\},N}}{G_{\mathcal{K},N}} = \lambda_d(0) \frac{G_{\mathcal{K} \setminus \{d\},N}}{G_{\mathcal{K},N}} ,$$

and then

$$X_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} X^c_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} \lambda_c(0)\frac{G_{\mathcal{K}\setminus\{c\},N}}{G_{\mathcal{K},N}} .$$

□

PROOF OF COROLLARY 3.9. Observe that $Y^d_{\mathcal{K},N} = \lambda_d(h_d)\pi^d_{\mathcal{K},N}(h_d)$ and $Y_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} Y^c_{\mathcal{K},N}$. By Lemma 3.5,

$$Y^d_{\mathcal{K},N} = \lambda_d(h_d)\pi^d_{\mathcal{K},N}(h_d) = \lambda_d(h_d)\binom{N}{h_d}\left(\prod_{j=0}^{h_d-1}\lambda_d(j)\right)\frac{G_{\mathcal{K}\setminus\{d\},N-h_d}}{G_{\mathcal{K},N}} ,$$

and then

$$Y_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} Y^c_{\mathcal{K},N} = \sum_{c\in\mathcal{K}} \lambda_c(h_c)\binom{N}{h_c}\prod_{j=0}^{h_c-1}\lambda_c(j)\frac{G_{\mathcal{K}\setminus\{c\},N-h_c}}{G_{\mathcal{K},N}} .$$

□

PROOF OF PROPOSITION 3.13. Max-min fairness is achieved when the set of λ^*_c is such that the increase of any rate must be at the cost of a decrease of some other already smaller (or equal) rate [Bertsekas and Gallager 1992]. Notice that, by definition, $\Phi_{\mathcal{K}_1} \geq 0$. Now, suppose that $\Phi_{\mathcal{K}_1} = 0$. Then, it means that for any pair $c, d \in \mathcal{K}_1$ either $\lambda_c - \lambda^*_c = 0$ or $\lambda^*_d - \lambda^*_c \leq 0$. Suppose that we can increase λ^*_c by decreasing λ^*_d with $\lambda^*_d > \lambda^*_c$. Then, clearly $\lambda_c - \lambda^*_c > 0$ and hence since $\Phi_{\mathcal{K}_1} = 0$ we must have $\lambda^*_d \leq \lambda^*_c$ which is a contradiction. Conversely, assume that λ^*_c, $c \in \mathcal{K}_1$, forms a max-min fair bandwidth assignment and suppose that $\Phi_{\mathcal{K}_1} > 0$. Then, there must exist at least one pair $c, d \in \mathcal{K}_1$ such that both $\lambda_c - \lambda^*_c > 0$ and $\lambda^*_d > \lambda^*_c$. Hence, we could increase the bandwidth assignment λ^*_c by decreasing λ^*_d which is a contradiction with the definition of max-min fairness. □

PROOF OF LEMMA 3.14. Let

$$\mathcal{S}_{\mathcal{K},N} = \{\mathbf{n} : \sum_{c\in\mathcal{K}} n_c = N \text{ and } 0 \leq n_c \leq h_c \ \forall c \in \mathcal{K}\} .$$

We first prove that $\mathcal{S}_{\mathcal{K},N}$ can be written as

$$\cup_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})}\{(\mathbf{n}_1,\mathbf{n}_2) :\ \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\} .$$

We distinguish the following four cases.

(1) Let $\tau_{\mathcal{K}_1}, \tau_{\mathcal{K}_2} \geq N$. In this case, it is easy to see that
$$\mathcal{S}_{\mathcal{K},N} = \cup_{j=0}^{N}\{(\mathbf{n}_1,\mathbf{n}_2) :\ \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\}$$
and then the thesis follows from the fact that $N = \min(N, \tau_{\mathcal{K}_2})$ and $0 = \max(0, N - \tau_{\mathcal{K}_1})$.

(2) Let $\tau_{\mathcal{K}_1} \geq N$ and $\tau_{\mathcal{K}_2} < N$. In this case, we have
$$\mathcal{S}_{\mathcal{K},N} = \cup_{j=0}^{\tau_{\mathcal{K}_2}}\{(\mathbf{n}_1,\mathbf{n}_2) :\ \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\}$$
and then the thesis follows from the fact that $\tau_{\mathcal{K}_2} = \min(N, \tau_{\mathcal{K}_2})$ and $0 = \max(0, N - \tau_{\mathcal{K}_1})$.

(3) Let $\tau_{\mathcal{K}_1} < N$ and $\tau_{\mathcal{K}_2} \geq N$. In this case, it holds that
$$\mathcal{S}_{\mathcal{K},N} = \cup_{j=N-\tau_{\mathcal{K}_1}}^{N}\{(\mathbf{n}_1,\mathbf{n}_2) : \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\}$$
and then the thesis follows from the fact that $N = \min(N, \tau_{\mathcal{K}_2})$ and $N - \tau_{\mathcal{K}_1} = \max(0, N - \tau_{\mathcal{K}_1})$.

(4) Let $\tau_{\mathcal{K}_1}, \tau_{\mathcal{K}_2} < N$. In this case, we have
$$\mathcal{S}_{\mathcal{K},N} = \cup_{j=N-\tau_{\mathcal{K}_1}}^{\tau_{\mathcal{K}_2}} \{(\mathbf{n}_1, \mathbf{n}_2) : \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\}$$
and then the thesis follows from the fact that $\tau_{\mathcal{K}_2} = \min(N, \tau_{\mathcal{K}_2})$ and $N - \tau_{\mathcal{K}_1} = \max(0, N - \tau_{\mathcal{K}_1})$.

Now by Corollary 3.4
$$G_{\mathcal{K},N} = \sum_{\mathbf{n}\in\mathcal{S}_{\mathcal{K},N}} \binom{N}{\mathbf{n}} \prod_{c\in\mathcal{K}} \prod_{j=0}^{n_c-1} \lambda_c(j) \,.$$
Assuming the classes $\mathcal{K}_1$ and $\mathcal{K}_2$ form a partition of $\mathcal{K}$, we can write
$$G_{\mathcal{K},N} = \sum_{\mathbf{n}\in\mathcal{S}_{\mathcal{K},N}} \binom{N}{\mathbf{n}} \prod_{c\in\mathcal{K}_1} \prod_{j=0}^{n_c-1} \lambda_c(j) \prod_{c\in\mathcal{K}_2} \prod_{j=0}^{n_c-1} \lambda_c(j)$$
that, from the fact that $\mathcal{S}_{\mathcal{K},N} = \cup_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \{(\mathbf{n}_1, \mathbf{n}_2) :\ \mathbf{n}_1 \in \mathcal{S}_{\mathcal{K}_1,N-j}, \mathbf{n}_2 \in \mathcal{S}_{\mathcal{K}_2,j}\}$, we have
$$G_{\mathcal{K},N} = \sum_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \sum_{\mathbf{n}_1\in\mathcal{S}_{\mathcal{K}_1,N-j},\ \mathbf{n}_2\in\mathcal{S}_{\mathcal{K}_2,j}} \binom{N}{\mathbf{n}_1\mathbf{n}_2} \prod_{c\in\mathcal{K}_1} \prod_{j=0}^{n_c-1} \lambda_c(j) \prod_{c\in\mathcal{K}_2} \prod_{j=0}^{n_c-1} \lambda_c(j) \,,$$
which is equal to
$$\sum_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \sum_{\mathbf{n}_1\in\mathcal{S}_{\mathcal{K}_1,N-j},\ \mathbf{n}_2\in\mathcal{S}_{\mathcal{K}_2,j}} \binom{N}{j} \binom{N-j}{\mathbf{n}_1} \binom{j}{\mathbf{n}_2} \prod_{c\in\mathcal{K}_1} \prod_{j=0}^{n_c-1} \lambda_c(j) \prod_{c\in\mathcal{K}_2} \prod_{j=0}^{n_c-1} \lambda_c(j) \,.$$
The latter can be written as
$$\sum_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \binom{N}{j} \sum_{\mathbf{n}_1\in\mathcal{S}_{\mathcal{K}_1,N-j}} \binom{N-j}{\mathbf{n}_1} \prod_{c\in\mathcal{K}_1} \prod_{j=1}^{n_c-1} \lambda_c(j) \sum_{\mathbf{n}_2\in\mathcal{S}_{\mathcal{K}_2,j}} \binom{j}{\mathbf{n}_2} \prod_{c\in\mathcal{K}_2} \prod_{j=1}^{n_c-1} \lambda_c(j)$$
and then we can write
$$G_{\mathcal{K},N} = \sum_{j=\max(0,N-\tau_{\mathcal{K}_1})}^{\min(N,\tau_{\mathcal{K}_2})} \binom{N}{j} G_{\mathcal{K}_1,N-j} G_{\mathcal{K}_2,j} \,.$$

□