Energy Packet Networks with Finite Capacity Energy Queues

Sébastien Samain

INRIA Paris
DI ENS, CNRS, PSL Research University
2, rue Simone Iff
Paris, France

Josu Doncel

University of the Basque Country
Barrio Sarriena s/n
Leioa, Spain

Ana Busic

INRIA Paris
DI ENS, CNRS, PSL Research University
2, rue Simone Iff
Paris, France

Jean-Michel Fourneau

DAVID, UVSQ, Univ. Paris Saclay
45, Avenue des Etats-Unis
Versailles, France

Abstract

Energy Packet Network (EPN) consists of a queueing network formed by N blocks, where each of them is formed by one data queue, that handles the workload, and one energy queue, that handles packets of energy.

We study an EPN model where the energy packets start the transfer. In this model, energy packets are sent to the data queue of the same block. An energy packet routes one workload packet to the next block if the data queue is not empty, and it is lost otherwise.

We assume that the energy queues have a finite buffer size and if an energy packet arrives to the system when the buffer is full, jump-over blocking (JOB) is performed, and therefore with some probability it is sent to the data queue and it is lost otherwise.

Email addresses: `sebastien.samain@inria.fr` (Sébastien Samain), `josu.doncel@ehu.eus` (Josu Doncel), `ana.busic@ehu.eus` (Ana Busic), `jean-michel.fourneau@uvsq.fr` (Jean-Michel Fourneau)

We first provide a value of the jump-over blocking probability such that the steady-state probability distribution of packets in the queues admits a product form solution. The product form is established for multiserver and multiclass data packet queues under FCFS, preemptive LCFS and PS discipline. Moreover, in the case of a directed tree queueing network, we show that the number of data packets in each subtree decreases as the JOB probability increases for each block.

1. Introduction

In the era of Internet of Things, Information and Communications Technology systems are growing at a very fast rate and, as a consequence, the performance analysis of such a huge network is a very challenging problem. Moreover, the source of energy that feeds this network includes an increasing amount of different types of renewable energies. The volatility of this kind of energy sources introduces clearly uncertainty in the amount of energy that is available in the future and, therefore, increases the difficulty of the optimal design of current communication systems.

Current technology allows energy to be stored in batteries or other devices so as to be used later. As a consequence, many researchers in the field of Computer Science have considered recently models where the energy is harvested. An example is the Energy Packet Network (EPN) model. This model has been introduced by Gelenbe and his colleagues [1] as a particular case of G-networks (we discuss in the related work section the literature on this topic). It considers that energy is represented by packets of discrete units of energy (Joules for instance) and, since its source is intermittent, it is assumed that arrivals to the system are given according to a random process. Therefore, in the EPN model, two types of packets are considered: on the one hand, the data packets that model the workload and are stored in the data queues; and on the other hand, the energy packets that are stored in the energy queues.

In this article, we study the EPN model where the energy packets start the transfer. This means that the energy packets are sent to the data queue and if, upon arrival, there is no data packet in the data queue, the energy packet is lost. However, if there are data packets available when the energy packet arrives, one data packet is sent to the next station and the energy packet disappears. This model captures well the performance of a system where tasks can only be executed when there is energy to feed the system. Sensor nodes and data centers are examples of these systems.

In the performance analysis literature, assumptions are sometimes considered that allow to get analytical results, but that are unrealistic from the practical point of view. This is the case, in fact, for most of the EPN models where the energy packet initiates the transfer, where it is considered that the energy queues (batteries) have a buffer of infinite size. In this article, we relax this assumption and we consider that the energy queues have a finite buffer size. We further assume that, if an energy packet arrives when the energy data is full, there is a jump-over blocking (JOB), which means that it is sent to the data queue with some probability and it is lost otherwise.

The main contributions of this article are summarized as follows:

- We analyze the stationary distribution of packets for an EPN model with multiclass data packets and the energy queues are multiserver queues. Moreover, when a data packet arrives to a data queue, it routes a job of a given class according to one of the following disciplines:

  - First Come First Served (FCFS)
  - Preemptive Last Come First Served (LCFS)
  - Processor Sharing (PS)

  and we show that both admit a product form solution for a given value of the probability at which an energy packet is sent to the data queue in case of jump-over blocking, i.e., in case of the energy queue is full.

- For queueing networks whose routing matrix is a directed tree, we show that there exists a stochastic ordering according to the probability at which an energy packet is sent to each data queue in case of jump-over blocking.

The rest of the article is organized as follows. In Section 2, we put our work in the context of the existing literature. We describe our model in Section 3. Then, in Section 4, we show the existence of the jump over blocking probabilities such that the distribution of packets in the queue admits a product form expression. We provide the stochastic ordering result in Section 5. Finally, we present the main conclusions of our work in Section 6.

A conference version of this article appeared in [2]. In fact, in [2] we studied a monoclass data packet multiserver model and in this article we consider multiclass data packets.

2. Related Work

The Energy Packet Network (EPN) models the interaction between intermittent sources of energy that come from batteries or renewal energy sources such as solar or wind and Information Technology devices that consume energy. This model was first studied in [3, 1, 4] by Gelenbe and his colleagues. It has attracted the attention of many researchers of different fields due to its wide range of applications in wireless sensors [5], mobile networks [6], computer systems design [7, 8], data centers [9] and optimization of power distribution policies [10, 11].

An interesting property of this model is that most of the EPN models that has been considered so far are particular cases of G-networks [12, 13, 14]. Since the steady-state distribution of packets in the queues of the G-networks is given by a product form expression, it is also the case for the EPN models, even for a general service time distribution [15]. As a result, one can study the performance of each node of the EPN network independently, which simplifies substantially the analysis of models with energy harvesting, see for instance [11] where the energy distribution of the EPN model is optimized. Let us note that all the EPN models are not always related to G-networks, see for instance, the model in [16] where the authors use a diffusion approximation to solve the interactions between Information Technology and energy.

All the EPN models that have been presented in the literature can be divided in two types depending on the initiator of the transfer (see [17] for a recent survey on EPN models). On the one hand, there are the models where the energy packets initiate the transfer (see for instance [7]). For this case, when the energy packets are sent to the data queue and are lost if there is no data packets. On the other hand, the data packets can start the transfer (see for example [18]), in which case the data packets are sent to the energy queue and are routed to the next data queue if there are energy packets and lost otherwise. We note that, in both cases, when a successful transfer occurs, the energy packet is removed from the system, whereas the data packet is sent to the next station or leaves the system.

In this work, we study EPNs in a network where the energy queue has a finite buffer size and the energy packets start the transfer. Thus, our EPN model extends the result of [19] of a single block and monoclass data packets to an arbitrary network with multiclass data packets. Moreover, our EPN model is different to [20], where the authors consider an EPN network formed by energy queues with infinite capacity and where the data packets start the transfer. Our model considers jump-over blocking when an energy packet arrives to an energy queue that is full. Therefore, this work is also clearly related to the literature of queueing-theory that applies this technique, see for instance [21, 22] for its application to Jackson Networks. We refer to [23] for full details about product-form results of queueing-theoretical models with finite buffer size.

3. Model Description

In this section, we present the model of Energy Packet Network that we study. The network consists of N stations or blocks, each of them formed by a data queue and an energy queue (or battery). There are K classes of data packets. The arrivals to the data queue of block i of a data packet of class l follow a Poisson process with rate $\lambda_i^{(l)}$. The arrivals to the energy queue of the data queue of block i are also Poisson with rate α_i. A leakage of an energy packet occurs with exponential time. We consider that the rate at which leakage of an energy packet of block i occurs is a function of the number of energy packets in that block, i.e., if there are y_i energy packets in block i, the leakage rate is denoted by $\beta_i(y_i)$.

In our model, energy packets start the transfer. This means that energy packets are sent to the data queue. We assume that the time required by an energy packet to reach the data queue is exponentially distributed with a rate that depends on the number of energy packets present in block i, that is, if there are y_i energy packets in the block i, this rate is denoted by $\mu_i(y_i)$. Note that this

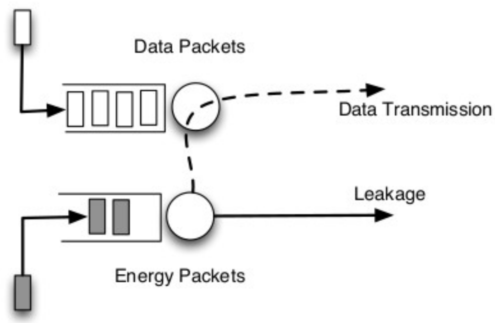

Figure 1: A single block of the EPN model under study.

rate does not depend on the number of data packets of block i. When an energy packet arrives to the data queue, it might find no data packets and, in this case, this energy is lost. However, when the data queue is not empty, we assume that a data packet of class l is routed to the next block (or it leaves the system) according to one of the following disciplines: FCFS, preemptive LCFS or PS. Under the FCFS discipline, the energy packet trigger the data packets in order of arrival. In the PS discipline, when the number of data packets of class l present in the data queue is z_l, a data packet of class l is routed to the next block (or leaves the system) with probability $\frac{z_l}{\sum_k z_k}$. We note that the discipline Random Order coincides with PS in our model. In the preemptive LCFS discipline, the energy packets trigger the data packet that has been waiting for the shortest time.

We assume that, in case of a successful transfer, a data packet of class l of block i is transmitted to the data queue of block j with probability $p_{i,j}^{(l)}$ and leaves the system with probability $p_{i,s}^{(l)}$. Therefore, for all i and all l, we have that

$$p_{i,s}^{(l)} + \sum_k p_{i,j}^{(l)} = 1.$$

A single block of the EPN model under consideration is shown in Figure 1.

We consider that the energy queue of block i has a finite buffer size, which we denote by B_i. If an energy packet arrives to block i when the energy queue of this block is full, jump-over blocking occurs. In other words, if the energy packet cannot be enqueued, it is either sent immediately to the data queue with jump-over blocking (JOB) probability q_i (where it is lost if the data queue is empty and, otherwise, it transfers a data packet) or lost with probability $1 - q_i$. We observe that, in both cases, the number of packets of the energy queue of block i does not change after the arrival of this energy packet, i.e., it remains full. Since in the EPN network under study some energy packets are loss, the designers of EPN networks are interested not only in standard performance metrics such as delay or number of customers in the queues, but also the number of loss energy packets.

The state of the network is represented by a couple of vectors (x, y), where $x = (x_1, \ldots, x_N)$ and $y = (y_1, \ldots, y_N)$. Thus, x_i represents the state of the data queue at node i and y_i represents the state of the battery at node i. The state of the energy queue of block i represents the number of energy packets in that block, which is an integer between 0 and B_i. The state of the data queue of block i depends on the discipline of that queue:

- for FCFS and LCFS, x_i is a word, where each letter is an integer number between 1 and K and the j-th letter of x_i, denoted by $r_{i,j}$, represents the class of the data packet at position j in the queue. Let us note that the length of the list represents the number of data packets in the data queue.

- for PS, x_i is a vector of K elements and the l-th element, $x_i^{(l)}$, represents the number of data packets of class l in the queue.

Remark 1. *We would like to remark that x_i and y_i are of different dimensions.*

We denote by $\pi(x, y)$ the steady state distribution of packets in the queues.

4. Product Form of the Steady State Distribution of Packets

In this section, we aim to study the steady state distribution of packets in the queues of the above described EPN model. For this purpose, we formulate the global balance equations. However, before writing down these equations, we need to introduce some notation to deal with the states, the rates and the functions taking into account the scheduling discipline at the data queue. We begin with the description of the states of the Markov chain in the multiple classes model. Note that $r_{i,k}$ is only defined for FCFS and LCFS discipline.

$$|x_i| = \begin{cases} \text{length of the list } x_i \text{ in the FCFS and LCFS case} \\ \sum_{l=1}^{K} x_i^{(l)} \text{ in the PS case} \end{cases}$$

$$r_{i,k} = \begin{cases} l \;\; \text{if the k-th letter of the word } x_i \text{ is } l \\ \text{undefined otherwise} \end{cases}$$

Following [20], we now define the following modifications of the state vectors:

- $x_i \oplus e^{(l)}$ is the state of block i such that, after a departure of a data packet of class l, the state of workstation i is x_i. Hence, $x_i \oplus e^{(l)}$ depends on the service discipline at block i:
  - *PS case:* if $x_i = (x_i^{(1)}, ..., x_i^{(K)})$, then $x_i \oplus e^{(l)}$ is obtained by adding 1 to the l-th component of x_i.

- *FCFS case:* if $x_i = [d_1, ..., d_{|x_i|}]$, with d_k being the class of the k-th data packet in the queue, then $x_i \oplus e^{(l)} = [l, d_1, ..., d_{|x_i|}]$.
- *LCFS case:* if $x_i = [d_1, ..., d_{|x_i|}]$, with d_k being the class of the k-th data packet in the queue, then $x_i \oplus e^{(l)} = [l, d_1, ..., d_{|x_i|}]$.

- $x \boxplus e_i^{(l)}$ is the state of the set of block, where the state of block i is replaced by $x_i \oplus e^{(l)}$.

- $x_i \ominus e^{(l)}$ is the state of block i such that, after an arrival of a data packet of class l, the state of block i is x_i, given that x_i represents a state of block i with at least one data packet of class l. Hence, $x_i \ominus e^{(l)}$ depends on the service discipline at block i:

  - *PS case:* if $x_i = (x_i^{(1)}, ..., x_i^{(K)})$, then $x_i \ominus e^{(l)}$ is obtained by subtracting 1 to the l-th component of x_i.
  - *FCFS case:* if $x_i = [d_1, ..., d_{|x_i|-1}, l]$, with d_k being the class of the k-th data packet in the queue, then $x_i \ominus e^{(l)} = [d_1, ..., d_{|x_i|-1}]$.
  - *LCFS case:* if $x_i = [l, d_2, ..., d_{|x_i|}]$, with d_k being the class of the k-th data packet in the queue, then $x_i \ominus e^{(l)} = [d_2, ..., d_{|x_i|}]$.

- $x \boxminus e_i^{(l)}$ is the state of the set of blocks, where the state of block i is replaced by $x_i \ominus e^{(l)}$.

and also the three functions below:

$$E_i^{(l)}(x_i \ominus e^{(l)}) = \begin{cases} 1_{\left[x_i^{(l)} > 0\right]} \text{ in the PS case} \\ 1_{\left[r_{i,|x_i|} = l\right]} \text{ in the FCFS case} \\ 1_{[r_{i,1} = l]} \text{ in the LCFS case} \end{cases}$$

$$M_i^{(l)}(x_i \oplus e^{(l)}) = \begin{cases} \frac{x_i^{(l)}+1}{|x_i|+1} \text{ in the PS case} \\ 1 \text{ in the FCFS or LCFS case} \end{cases}$$

$$L_i^{(l)}(x_i \ominus e^{(l)}) = \begin{cases} \frac{x_i^{(l)}}{|x_i|} 1_{[|x_i|>0]} \text{ in the PS case} \\ 1_{\left[r_{i,|x_i|} = l\right]} \text{ in the FCFS case} \\ 1_{[r_{i,1} = l]} \text{ in the LCFS case} \end{cases}$$

The first one indicates whether the state $x_i \ominus e^{(l)}$ can be defined. The second one indicates the proportion of the service rate devoted to the service of the class l client served at the end of the service time. The third one is equal to $E_i^{(l)}(x_i \ominus e^{(l)})$ multiplied by $\frac{x_i^{(l)}}{|x_i|}$ is the PS case, and by 1 otherwise.

Now, we can write down the Kolmogorov equation at steady-state:

$$\pi(x,y)\left(\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}+\alpha_i 1_{[y_i<B_i]}+\left(\beta_i(y_i)+\mu_i(y_i)\right)1_{[y_i>0]}+\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}\right)\right)=$$

$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}\pi(x\boxminus e_i^{(l)},y)E_i^{(l)}(x_i\ominus e^{(l)})+\alpha_i\pi(x,y-e_i)1_{[y_i>0]}\right.$$
$$+\beta_i(y_i+1)\pi(x,y+e_i)1_{[y_i<B_i]}+\mu_i(y_i+1)\pi(x,y+e_i)1_{[|x_i|=0,y_i<B_i]}$$
$$+\sum_{j=1}^{N}\sum_{l=1}^{K}\mu_i(y_i+1)\pi(x\boxplus e_i^{(l)}\boxminus e_j^{(l)},y+e_i)p_{i,j}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j^{(l)}(x_j\ominus e^{(l)})$$
$$+\sum_{j=1}^{N}\sum_{l=1}^{K}\alpha_i q_i\pi(x\boxplus e_i^{(l)}\boxminus e_j^{(l)},y)p_{i,j}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j^{(l)}(x_j\ominus e^{(l)})$$
$$+\sum_{l=1}^{K}\mu_i(y_i+1)\pi(x\boxplus e_i^{(l)},y+e_i)p_{i,s}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})$$
$$\left.+\sum_{l=1}^{K}\alpha_i q_i\pi(x\boxplus e_i^{(l)},y)p_{i,s}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})\right)$$

In the LHS of the above expression, we represent the total flow out from state (x,y). The RHS consists of 6 lines and it is formed by the total flow into (x,y). In the first line, we represent the flow due to an arriving data packet, an arriving energy packet and the leakage of an energy packet. In the second line, we represent the flow of an energy packet going to an empty data queue. In the following two lines, we represent the flow of an data packet routed to another queue after receiving an energy packet: in the first one, the energy packet goes to the data queue after being served, and in the second one the energy packet goes to the data queue by jump-over blocking. Finally, the last two lines show the flow of a data packet that leaves the system: in the first one, the energy packet goes to the data queue after being served, and in the second one the energy packet goes to the data queue by jump-over blocking.

We consider there exists a function $f_i : [\![0,B_i]\!] \to [\![1,B_i]\!]$ and b_i and m_i a pair of positive constants (i.e., they do not depend on the state y_i) such that $\beta_i(y_i) = b_i f_i(y_i)$ and $\mu_i(y_i) = m_i f_i(y_i)$. In the following result, we show that the steady-state distribution of packets in the queues verifies a product form expression when the jump-over blocking probability is $q_i = \frac{m_i}{m_i+b_i}$.

Theorem 1. *Assume that the Markov chain modeling the network is ergodic. Let $q_i = \frac{m_i}{m_i+b_i}$. Consider the flow equations for the data packets:*

$$\rho_i^{(l)} = \frac{\lambda_i^{(l)} + \sum_{j=1}^{N}\alpha_j q_j \rho_j^{(l)} p_{j,i}^{(l)}}{\alpha_i q_i}. \tag{1}$$

If these equations have a solution such that

$$\forall (i,l) \in [\![1,N]\!]\times[\![1,K]\!], \rho_i^{(l)} > 0,$$

$$\forall i \in [\![1, N]\!], \sum_{l=1}^{K} \rho_i^{(l)} < 1,$$

then the steady-state distribution has a product form:

$$\pi(x, y) = \left(\prod_{i=1}^{N} C_i \left(1 - \sum_{l=1}^{K} \rho_i^{(l)} \right) g_i(x_i) \prod_{k=1}^{y_i} \gamma_i(k) \right)$$

where the flow equation of energy packets is

$$\gamma_i(y_i) = \frac{\alpha_i}{\beta_i(y_i) + \mu_i(y_i)}, \tag{2}$$

and

$$g_i(x_i) = \begin{cases} |x_i|! \prod_{l=1}^{K} \frac{\left(\rho_i^{(l)}\right)^{x_i^{(l)}}}{x_i^{(l)}!} & \textsf{\textit{in the PS case}} \\ \prod_{j=1}^{|x_i|} \rho_i^{(r_{i,j})} & \textsf{\textit{in the FCFS or LCFS case.}} \end{cases}$$

Moreover, the normalization constant is given by

$$C_i = \frac{1}{\sum_{j=0}^{B_i} \prod_{k=1}^{j} \gamma_i(k)}$$

The proof of this theorem is also based on the manipulation of the global balance equation. As these manipulations are rather technical, we first give some lemmas to show how we can deal with the scheduling disciplines. The proofs of the lemmas are postponed in an appendix for the sake of readability. We first remark that combining the constraint on the jump-over probability, and the energy packet flow equation we have:

$$\mu_i(y_i)\gamma_i(y_i) = \frac{\mu_i(y_i)\alpha_i}{\mu_i(y_i) + \beta_i(y_i)} = \alpha_i q_i.$$

Similarly, we get $\beta_i(y_i)\gamma_i(y_i) = \alpha_i(1 - q_i)$. Both relations will be used during the analysis of the global balance equations.

Lemma 1. *For all $i \in [\![1, N]\!]$, we have that*

$$\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} M_i^{(l)}(x_i \oplus e^{(l)}) = \rho_i^{(l)}$$

Lemma 2. *For all $i \in [\![1, N]\!]$, we have that*

$$\frac{g_i(x_i \ominus e^{(l)})}{g_i(x_i)} E_i^{(l)}(x_i \ominus e^{(l)}) = \frac{1}{\rho_i^{(l)}} L_i^{(l)}(x_i \ominus e^{(l)})$$

Lemma 3. *For all $i \in [\![1, N]\!]$, we have that*

$$1_{[|x_i|>0]} = \sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)})$$

Lemma 4. *The following relations hold for all $i \in [\![1, N]\!]$:*

$$\sum_{j=1}^{N}\sum_{l=1}^{K} \mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j \ominus e^{(l)})}{g_j(x_j)} p_{i,j}^{(l)} 1_{[y_i<B_i]} M_i^{(l)}(x_i \oplus e^{(l)}) E_j(x_j \ominus e^{(l)})$$
$$+ \sum_{j=1}^{N}\sum_{l=1}^{K} \alpha_i q_i \frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j \ominus e^{(l)})}{g_j(x_j)} p_{i,j}^{(l)} 1_{[y_i=B_i]} M_i^{(l)}(x_i \oplus e^{(l)}) E_j(x_j \ominus e^{(l)})$$
$$= \sum_{j=1}^{N}\sum_{l=1}^{K} \frac{\alpha_i q_i \rho_i^{(l)}}{\rho_j^{(l)}} p_{i,j}^{(l)} L_j(x_j \ominus e^{(l)}),$$

and

$$\sum_{l=1}^{K} \mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} p_{i,s}^{(l)} 1_{[y_i<B_i]} M_i^{(l)}(x_i \oplus e^{(l)})$$
$$+ \sum_{l=1}^{K} \alpha_i q_i \frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} p_{i,s}^{(l)} 1_{[y_i=B_i]} M_i^{(l)}(x_i \oplus e^{(l)})$$
$$= \sum_{l=1}^{K} \alpha_i q_i \rho_i^{(l)} p_{i,s}^{(l)}$$

Lemma 5. *The flow equation between the network and the outside holds as a consequence of the flow equation of the data packet for each queue:*

$$\sum_{i=1}^{N}\sum_{l=1}^{K} \alpha_i q_i \rho_i^{(l)} p_{i,s}^{(l)} = \sum_{l=1}^{K}\sum_{i=1}^{N} \lambda_i^{(l)}$$

Lemma 6. *π is a probability distribution.*

Now, we can turn back to the proof of Theorem 1:

Proof of Theorem 1. By Lemma 6, π is a probability distribution; hence we need to check that it satisfies the global balance equation and we are done.

We now divide both sides of the global balance equation by $\pi(x, y)$, and we use the multiplicative solution to get:

$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}+\alpha_i 1_{[y_i<B_i]}+\left(\beta_i(y_i)+\mu_i(y_i)\right)1_{[y_i>0]}+\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}\right)=$$

$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}\frac{g_i(x_i\ominus e^{(l)})}{g_i(x_i)}E_i^{(l)}(x_i\ominus e^{(l)})+\frac{\alpha_i}{\gamma_i(y_i)}1_{[y_i>0]}+\beta_i(y_i+1)\gamma_i(y_i+1)1_{[y_i<B_i]}\right.$$

$$+\mu_i(y_i+1)\gamma_i(y_i+1)1_{[|x_i|=0,y_i<B_i]}$$

$$+\sum_{j=1}^{N}\sum_{l=1}^{K}\mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j^{(l)}(x_j\ominus e^{(l)})$$

$$+\sum_{j=1}^{N}\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j^{(l)}(x_j\ominus e^{(l)})$$

$$+\sum_{l=1}^{K}\mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})$$

$$\left.+\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})\right)$$

Now we have several simplifications which are based on the lemmas and the assumptions:

- From Lemma 2, we get :

$$\sum_{l=1}^{K}\lambda_i^{(l)}\frac{g_i(x_i\ominus e^{(l)})}{g_i(x_i)}E_i^{(l)}(x_i\ominus e^{(l)})=\sum_{l=1}^{K}\frac{\lambda_i^{(l)}}{\rho_i^{(l)}}L_i^{(l)}(x_i\ominus e^{(l)})$$

- Due to the energy packets flow equation: $\beta_i(y_i)+\mu_i(y_i)=\frac{\alpha_i}{\gamma_i(y_i)}$, the term $\left(\beta_i(y_i)+\mu_i(y_i)\right)1_{[y_i>0]}$ on the LHS cancels with the term $\frac{\alpha_i}{\gamma_i(y_i)}1_{[y_i>0]}$ on the RHS .

- Combining the constraint on the jump-over probability, and the energy flow equation we obtain:

$$\beta_i(y_i+1)\gamma_i(y_i+1)1_{[y_i<B_i]}+\mu_i(y_i+1)\gamma_i(y_i+1)1_{[|x_i|=0,y_i<B_i]}=$$
$$\alpha_i(1-q_i)1_{[y_i<B_i]}+\alpha_i q_i 1_{[|x_i|=0,y_i<B_i]}$$

- We also apply Lemmas 1, 2 and 4

After substitution, we obtain:

$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}+\alpha_i 1_{[y_i<B_i]}+\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}\right)=$$
$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\frac{\lambda_i^{(l)}}{\rho_i^{(l)}}L_i^{(l)}(x_i\ominus e^{(l)})+\alpha_i(1-q_i)1_{[y_i<B_i]}+\alpha_i q_i 1_{[|x_i|=0,y_i<B_i]}\right.$$
$$\left.+\sum_{j=1}^{N}\sum_{l=1}^{K}\frac{\alpha_i q_i\rho_i^{(l)}}{\rho_j^{(l)}}p_{i,j}^{(l)}L_j(x_j\ominus e^{(l)})+\sum_{l=1}^{K}\alpha_i q_i\rho_i^{(l)}p_{i,s}^{(l)}\right)$$

Let us now consider all the terms which contains a indicator function associated with B_i and regroup them. We have:

$$\begin{aligned}
&\alpha_i 1_{[y_i<B_i]}+\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}-\left(\alpha_i(1-q_i)1_{[y_i<B_i]}+\alpha_i q_i 1_{[|x_i|=0,y_i<B_i]}\right)\\
&=\alpha_i 1_{[y_i<B_i]}+\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}-\alpha_i 1_{[y_i<B_i]}+\alpha_i q_i 1_{[y_i<B_i]}-\alpha_i q_i 1_{[|x_i|=0,y_i<B_i]}\\
&=\alpha_i q_i 1_{[y_i=B_i,|x_i|>0]}+\alpha_i q_i 1_{[y_i<B_i]}-\alpha_i q_i 1_{[|x_i|=0,y_i<B_i]}\\
&=\alpha_i q_i\left(1_{[y_i=B_i,|x_i|>0]}+1_{[y_i<B_i]}-1_{[|x_i|=0,y_i<B_i]}\right)\\
&=\alpha_i q_i\left(1_{[y_i=B_i,|x_i|>0]}+1_{[y_i<B_i,|x_i|>0]}\right)\\
&=\alpha_i q_i 1_{[|x_i|>0]}
\end{aligned}$$

Hence, we substitute the result in the Kolmogorov equation and we get:

$$\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\lambda_i^{(l)}+\alpha_i q_i 1_{[|x_i|>0]}\right)=\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\frac{\lambda_i^{(l)}}{\rho_i^{(l)}}L_i^{(l)}(x_i\ominus e^{(l)})\right.$$
$$\left.+\sum_{j=1}^{N}\sum_{l=1}^{K}\frac{\alpha_i q_i\rho_i^{(l)}}{\rho_j^{(l)}}p_{i,j}^{(l)}L_j(x_j\ominus e^{(l)})+\sum_{l=1}^{K}\alpha_i q_i\rho_i^{(l)}p_{i,s}^{(l)}\right)$$

We clearly have two equations which are combined: one of them using explicitly or implicitly (via $L_i^{(l)}$) an indicator function on x_i while the other one does not have state dependent terms.

$$\sum_{i=1}^{N}\alpha_i q_i 1_{[|x_i|>0]}=\sum_{i=1}^{N}\left(\sum_{l=1}^{K}\frac{\lambda_i^{(l)}}{\rho_i^{(l)}}L_i^{(l)}(x_i\ominus e^{(l)})+\sum_{j=1}^{N}\sum_{l=1}^{K}\frac{\alpha_i q_i\rho_i^{(l)}}{\rho_j^{(l)}}p_{i,j}^{(l)}L_j(x_j\ominus e^{(l)})\right) \tag{3}$$

and

$$\sum_{i=1}^{N}\sum_{l=1}^{K}\lambda_i^{(l)}=\sum_{l=1}^{K}\alpha_i q_i\rho_i^{(l)}p_{i,s}^{(l)} \tag{4}$$

This last equation holds. It is the flow equation between the network and the outside. It has been established in Lemma 5. Therefore it remains to prove that

(3) holds. We exchange indices i and j in the second term of the RHS of (3) and we factorize the RHS.

$$\sum_{i=1}^{N} \alpha_i q_i 1_{[|x_i|>0]} = \sum_{i=1}^{N} \sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)}) (\frac{\lambda_i^{(l)}}{\rho_i^{(l)}} + \sum_{j=1}^{N} \frac{\alpha_j q_j \rho_j^{(l)}}{\rho_i^{(l)}} p_{j,i}^{(l)})$$

Due to the flow equation for the data packets (i.e. (1)), we have for all l:

$$\frac{\lambda_i^{(l)}}{\rho_i^{(l)}} + \sum_{j=1}^{N} \frac{\alpha_j q_j \rho_j^{(l)}}{\rho_i^{(l)}} p_{j,i}^{(l)} = \alpha_i q_i$$

And Lemma 3 states that $\sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)}) = 1_{[|x_i|>0]}$ for all queues irrespective of their service discipline. As a consequence, (3) holds. Therefore, the desired result follows. $\square$

We can then obtain a more compact formulation of the probability distribution by a summation on the state space for queues with FCFS or LCFS discipline.

Corollary 1. *Under the assumptions of the previous theorem, we have a simplified formulation for the join probability of the number of customers in a queue irrespective of the service discipline:*

$$Pr(x, y) = \prod_{i=1}^{N} (1 - \sum_{l=1}^{K} \rho_i^{(l)}) |x_i|! \prod_{l=1}^{K} \frac{\left(\rho_i^{(l)}\right)^{x_i^{(l)}}}{x_i^{(l)}!} \prod_{j=1}^{N} C_j \gamma_j(y_j)$$

where $x_i^{(l)}$ is the number of class l data packet in queue i.

Proof. These probabilities are obtained by summation on the elementary probabilities obtained in the previous theorem. Indeed, in a FCFS or a LCFS queue, all the states with the same number $x_i^{(l)}$ have the same probability due to the multiplicative form of the solution and the multinomial coefficients appear as usual in such a summation. $\square$

We would like to remark that our result covers a wide range of cases of interest. For instance, using the above results, we conclude the existence of a product-form expression when the energy queues are $M/M/1/B_i$ queues. Indeed, for this case, we have that $\mu_i(y_i) = \mu_i$ and $\beta_i(y_i) = \beta_i$, which satisfy the condition of our theorem. Furthermore, the existence of the product form for energy queues that are $M/M/B_i/B_i$ queues also follows from the above results since for that case, we have that $\mu_i(y_i) = y_i \mu_i$ and $\beta_i(y_i) = y_i \beta_i$.
We also remark that the value of ρ_i obtained in the result above does not depend on the value of the buffer size B_i and coincides with that of the corresponding EPN model with infinite capacity. The main reason for this is the way that the jump-over blocking is performed in our model. Besides, this property means

that themodel with infinite capacity and our model coincide in performance metrics of interest such as the mean number of data packets. However, we remark that, in our model, the stability of energy packets is not an issue, whereas in the infinite capacity packets it must besatisfied that $\gamma_i(y_i) < 1$.

5. Stochastic ordering

In this section, we focus on a single block and we study the influence of the probability that an energy packet is sent to the data queue when the jump-over blocking occurs (i.e., when the energy queue is full). Since we consider a single block, we drop the subindex i of the parameters of the system in this part of the article.

We first define a partial order $\preceq_S$ on the state space S of our model. We say that $(x_1, y_1) \preceq_S (x_2, y_2)$ if $x_1 \le x_2$ and $y_1 \ge y_2$. The intuitive idea of this ordering is that it is preferable to have (i) less data packets and (ii) more energy packets to be sent to the data queue and to route them to the next station. In other words, the energy packets play the role of the servers for the data queues.

We wish to compare the continuous time Markov chains (CTMC) corresponding to two single blocks with different values of JOB probabilities q according to the strong stochastic order. We refer to Appendix B for the definition of the strong stochastic ordering as well as the properties we require to show the desired result.

We now present the main result of this section:

Theorem 2. *Consider an EPN network with a single block, when the energy queue is a M/M/1/B queue or a M/M/B/B queue; if Z is a CTMC of this model with JOB probability q and Z' is a CTMC of this model with JOB probability q' such that $q \le q'$, then*

$$Z'_t \preceq_{st} Z_t.$$

Corollary 2. *The model with full rejection of the energy packet (i.e. $q = 0$) is greater (in the strong stochastic sense) than the model with jump-over blocking ($q > 0$).*

We now turn to the proof of theorem 2. We prove that Corollary 3 (see Appendix B) holds for one-block EPN by using the following event representation (below, $z = (x, y)$):

- $a1$: *arrival of an energy packet of type 1* (jump-over blocking performed when the energy queue is full) with rate $\tau_{a1}(z) = q\alpha$;
- $a2$: *arrival of an energy packet of type 2* (jump-over blocking not performed when the energy queue is full) with rate $\tau_{a2}(z) = (1 - q)\alpha$;
- d: *arrival of a data packet* with rate $\tau_d(z) = \lambda$;
- b: *leakage of an energy packet* with rate $\tau_b(z) = \beta(y)$;

- s: *service of a data packet, triggered by an energy packet* with rate $\tau_s(z) = \mu(y)$.

The reason to split arrivals of energy packets into two types of events is purely to be able to describe the effect of the event by deterministic functions. To each event e, we associate a function $t_e : S \to S$ defined as follows:

- $a1$: $t_{a1}(x, y) = (x, y + 1)1_{[y<B]} + ((x - 1)^+, y)1_{[y=B]}$;
- $a2$: $t_{a2}(x, y) = (x, y + 1)1_{[y<B]} + (x, y)1_{[y=B]}$;
- d: $t_d(x, y) = (x + 1, y)$;
- b: $t_b(x, y) = (x, (y - 1)^+)$;
- s: $t_s(x, y) = ((x - 1)^+, y - 1)1_{[y>0]} + (x, y)1_{[y=0]}$,

and a generator $Q_e = \Delta(\tau_e)(E(t_e) - I)$, with $\Delta(\tau_e)$ the diagonal matrix of rates and $E(t_e) = (1_{[t(z_1)=z_2]})_{(z_1,z_2)\in S\times S}$. Let $q \in [0, 1]$ and $Q_{a,q} = Q_{a1} + Q_{a2}$ where the rate function for the events $a1$ and $a2$ are respectively αq and $\alpha(1 - q)$. If Q is the generator of the CTMC associated to the one-block EPN with JOB probability q, then we have $Q = Q_d + Q_{a,q} + Q_b + Q_s$.

An event e is said to be *st*-monotone if its generator Q_e is *st*-monotone. If the generator Q of a CTMC can be written $Q = \sum_{e\in\mathcal{E}} Q_e$ such that every $e \in \mathcal{E}$ is *st*-monotone, then Q *st*-monotone [24]. Hence, we want to prove that every $e \in \{a1, a2, d, b, s\}$ is st-monotone. To this aim, we will use the following result from [24] that characterizes the *st*-monotonicity of an event:

Theorem 3 ([24, Thm 5.4]). *Let e be an event with destination $t : S \to S$ and rate $\tau : S \to \mathbb{R}^+$. Event e is st-monotone if and only if the following conditions are verified for all $z_1 = (x_1, y_1), z_2 = (x_2, y_2)$ such that $z_1 \preceq_S z_2$:*

1) If $\tau(z_1)$ and $\tau(z_2)$ are nonzero, then at least one of the conditions must hold:

   a) $t(z_1) \preceq_S t(z_2)$,

   b) $z_1 \preceq_S t(z_2)$ and $t(z_1) \preceq_S z_2$.

2) If $\tau(z_1) < \tau(z_2)$, then $z_1 \preceq_S t(z_2)$.

3) If $\tau(z_1) > \tau(z_2)$, then $t(z_1) \preceq_S z_2$.

In the two following lemmas, we show that all the events $e \in \{a1, a2, d, b, s\}$ are st-monotone for the cases where energy queue is a M/M/1/B queue or a M/M/B/B queue. We deal first with the M/M/1/B case:

Lemma 7. *Let $e \in \{a1, a2, b, d, s\}$ in a EPN with a single block, when the energy queue is an M/M/1/B queue. Then, e is st-monotone.*

Proof. All the rates are state-independent, i.e., $\tau_e(z)$ is a constant that only depends on $e \in \{a1, a2, b, d, s\}$. Therefore, conditions 2) and 3) of Theorem 3 are never verified and, as a consequence, to show that an event e is st-monotone, it is enough to show that condition 1) a) of Theorem 3 is satisfied.

To show this condition, for each event e, we partition the state space S into "types" of states such that t_e is a translation with respect to $\preceq_S$ on each "type set" (a subset of states of one given type), that is, if (a) is a type of states for e and $A_{(a)}$ is the subset of states of type (a), then $\forall z \in A_{(a)}, t_e(z) = z + v_{(a)}$, with $v_{(a)}$ a vector which depends only on the type (a); hence, on each of type sets, condition 1) a) will hold; moreover, when considering couples $(z_1, z_2) \in S^2$ such that $z_1 \preceq z_2$ and the pair (z_1, z_2) covers two different types (that is, z_1 is not of the same type as z_2), some cases will be forbidden; for instance, if a type (a) requires $z = (x, y)$ to be such that $x = 0$ and a type (b) requires $z = (x, y)$ to be such that $x > 0$, then we can have z_1 of type (a) and z_2 of type (b), but not the converse; similarly, if a type (a) requires $z = (x, y)$ to be such that $y = B$ and a type (b) requires $z = (x, y)$ to be such that $y < B$, then we can have z_1 of type (a) and z_2 of type (b), but not the converse; finally, if a type (a) requires $z = (x, y)$ to be such that $y > 0$ and a type (b) requires $z = (x, y)$ to be such that $y = 0$, then we can have z_1 of type (a) and z_2 of type (b), but not the converse.

In the following, it will be assumed that the states z, z_1 and z_2 can be written as $(x, y), (x_1, y_1)$ and (x_2, y_2) respectively. We examine each type of event and show that they are all st-monotone.

<u>Event d is st-monotone.</u> If $z_1 \preceq_S z_2$, we have $t_d(z_1) = (x_1 + 1, y_1) \preceq_S (x_2 + 1, y_2) = t_d(z_2)$. Hence, condition 1) a) holds.

<u>Event b is st-monotone.</u> For $z \in S$, we distinguish two types: (i) $y > 0$ and (ii) $y = 0$. Let $(z_1, z_2) \in S^2$ such that $z_1 \preceq_S z_2$. If z_1 and z_2 have same type then as t_b is a translation by $(0, -1)$ on $A_{(i)}$ and by $(0, 0)$ on $A_{(ii)}$, condition 1) a) holds when z_1 and z_2 have the same type. If z_1 and z_2 cover types (i) and (ii), then z_1 is of type (i) and z_2 is of type (ii), and $t_b(z_1) = (x_1, y_1 - 1) \preceq_S (x_2, 0) = t_b(z_2)$. Hence, condition 1) a) holds.

<u>Event s is st-monotone.</u> For $z \in S$, we distinguish three types: (i) $y > 0, x > 0$, (ii) $y > 0, x = 0$ and (iii) $y = 0$. Let $(z_1, z_2) \in S^2$ such that $z_1 \preceq_S z_2$. As t_s is a translation by $(-1, -1)$ on $A_{(i)}$, by $(0, -1)$ on $A_{(ii)}$ and by $(0, 0)$ on $A_{(iii)}$, condition 1) a) holds when z_1 and z_2 have the same type. If z_1 and z_2 cover types (i) and (ii), then z_1 is of type (ii), z_2 is of type (i) and we have $t_s(z_1) = (0, y_1 - 1) \preceq_S (x_2 - 1, y_2 - 1) = t_s(z_2)$. If z_1 and z_2 cover types (i) and (iii), then z_1 is of type (i), z_2 is of type (iii) and $t_s(z_1) = (x_1 - 1, y_1 - 1) \preceq_S (x_2 - 1, 0) \preceq_S (x_2, 0) = t_s(z_2)$. Finally, if z_1 and z_2 cover types (ii) and (iii) then z_1 is of type (ii) and z_2 is of type (iii), and $t_s(z_1) = (0, y_1 - 1) \preceq_S (x_2, 0) = t(z_2)$. Hence, condition 1) a) holds.

<u>Event $a2$ is st-monotone.</u> For $z \in S$, we distinguish two types: (i) $y < B$ and (ii) $y = B$. Let $(z_1, z_2) \in S^2$ such that $z_1 \preceq_S z_2$. As t_{a2} is a translation by $(0, 1)$ on $A_{(i)}$ and by $(0, 0)$ on $A_{(ii)}$, condition 1) a) holds when z_1 and z_2 have the same type. If z_1 and z_2 cover types (i) and (ii), then z_1 is of type (ii), z_2 is of type (i) and $t_{a2}(z_1) = (x_1, B) \preceq_S (x_2, y_2 + 1) = t_{a2}(z_2)$. Hence, condition 1)

a) holds.

Event $a1$ is st-monotone. For $z \in S$, we distinguish three types: (i) $y < B$, (ii) $y = B, x > 0$ and (iii) $y = B, x = 0$. Let $(z_1, z_2) \in S^2$ such that $z_1 \preceq_S z_2$; as t_{a_1} is a translation by $(0, 1)$ on $A_{(i)}$, by $(-1, -1)$ on $A_{(ii)}$ and by $(0, 0)$ on $A_{(iii)}$, condition 1) a) holds when z_1 and z_2 have the same type. If z_1 and z_2 cover types (i) and (ii), then z_1 is of type (ii), z_2 is of type (i) and $t_{a_1}(z_1) = (x_1 - 1, B) \preceq_S (x_1, B) \preceq_S (x_2, B) \preceq (x_2, y_2 + 1) = t_{a1}(z_2)$. If z_1 and z_2 cover types (i) and (iii), then z_1 is of type (iii), z_2 is of type (i) and $t_{a1}(z_1) = (0, B) \preceq_S (x_2, y_2 + 1) = t_{a2}(z_2)$. Finally, if z_1 and z_2 cover types (ii) and type (iii), then z_1 is of type (iii), z_2 is of type (ii) and $t_{a1}(z_1) \preceq_S (0, B) \preceq_S (x_2 - 1, B) = t_{a1}(z_2)$. Hence, condition 1) a) holds. □

We now consider that the energy queues are M/M/B/B queues.

Lemma 8. *Let $e \in \{a1, a2, b, d, s\}$ in a EPN with a single block, when the energy queue is an M/M/B/B queue. Then, e is st-monotone.*

Proof. Using the same arguments as in the energy queue that is a M/M/1/B queue, we can easily show that events $a1$, $a2$ and d are st-monotone for this case as well.

The case of events b and s is different, because we have $\beta(y) = y\beta$ and $\mu(y) = y\mu$; hence, in addition to condition 1) a), we must also verify condition 3) for these events. The proof of condition 1) a) is the same as in the proof of lemma 7. Hence, we focus only on condition 3).

We show that the event b is st-monotone. For $z \in S$, we distinguish two types: $(i) y > 0$ and $(ii) y = 0$ Let $z_1 = (x_1, y_1), z_2 = (x_2, y_2) \in S$ such that $z_1 \preceq_S z_2$. If $y_1 = y_2$, then $\beta(y_1) = \beta(y_2)$ and we need not to verify condition 3). If $y_1 > y_2$, then $\beta(y_1) > \beta(y_2)$ and therefore, we need to verify that condition 3) of Theorem 3 is satisfied. In this case, z_1 must be of type (i) and z_2 can be either of type (i) or (ii). In both cases, we have $t_b(z_1) = (x_1, y_1 - 1) = (x_1, y_2) \preceq_S z_2$, and hence, condition 3) holds.

We now show that the event s is st-monotone. For $z \in S$, we distinguish three types:

- (i) states z for which $y > 0, x > 0$
- (ii) states z for which $y > 0, x = 0$
- (iii) states z for which $y = 0$

Let $z_1 = (x_1, y_1), z_2 = (x_2, y_2) \in S$ such that $z_1 \preceq_S z_2$. If $y_1 = y_2$, then $\mu(y_1) = \mu(y_2)$ and we need not to verify condition 3). If $y_1 \neq y_2$, then $y_1 > y_2$ and $\mu(y_1) > \mu(y_2)$. Therefore, we need to verify that condition 3) of Theorem 3 is satisfied. We distinguish the following cases:

- z_1 is of type (i): as $y_1 - 1 \geq y_2$, we have $t_s(z_1) = (x_1 - 1, y_1 - 1) \preceq_S (x_2, y_2) = z_2$, and hence condition 3) holds.

- z_1 is of type (ii): we have $t_s(z_1) = (x_1, y_1 - 1) \preceq_S (x_2, y_2) = z_2$, and hence condition 3) holds.

- z_1 is of type (iii): in this case, we also have z_2 of type (iii), thus the service rate is zero for both z_1 and z_2. Hence we do not need to prove condition 3).

Hence, condition 3) holds for the event s.

□

We thus have proved, by summing the generators of the events, that the generator Q_q of the CTMC associated with the one-block EPN model with JOB probability q is st-monotone. We now want to show that when $(q, q') \in [0, 1]^2$ are such that $q' \geq q$, then $Q_{q'} \preceq_{st} Q_q$. Again, our event representation is will help us, due to the following lemma:

Lemma 9. *Let $\mathcal{E}$ be a set of events and for any $e \in \mathcal{E}$, let Q_e and R_e be generators with R_e $\preceq_{st}$-monotone such that $Q_e \preceq_{st} R_e$, then $\sum_{e\in\mathcal{E}} Q_e \preceq_{st} \sum_{e\in\mathcal{E}} R_e$.*

As Q_e is independent of the JOB probability for any $e \in \{d, b, s\}$, and Q_e is st-monotone for any $e \in \{a1, a2, d, b, s\}$ when the energy queue is either M/M/1/B or M/M/B/B, we only need to show the following lemma:

Lemma 10. *If $q' > q$, then $Q_{a,q'} \preceq_{st} Q_{a,q}$.*

Proof. By definition 4, we need to show that for any $z = (x, y) \in S$ and any increasing set $\Gamma \subseteq S$, we have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) \leq \sum_{w\in\Gamma} Q_{a,q}(z, w)$. To this aim, we distinguish two cases:

- $y < B$ or $z = (0, B)$: in this case, we have that $Q_{a,q}(z, .) = Q_{a,q'}(z, .)$, and we immediately have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) \leq \sum_{w\in\Gamma} Q_{a,q}(z, w)$.

- $y = B$ and $x > 0$: in this case, as we have $t_{a1}(z) \preceq_S z$ and Γ is an increasing set, we only need to distinguish three cases:
  - $z \notin \Gamma$: in this case, $t_{a1}(z) \notin \Gamma$ and we necessarily have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) = 0$ and $\sum_{w\in\Gamma} Q_{a,q}(z, w) = 0$.
  - $z \in \Gamma, t_{a1}(z) \notin \Gamma$: in this case, we have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) = -\alpha q' \leq -\alpha q = \sum_{w\in\Gamma} Q_{a,q}(z, w)$.
  - $z \in \Gamma, t_{a1}(z) \notin \Gamma$: in this case, we have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) = Q_{a,q'}(z, z) + Q_{a,q'}(z, t_{a1}(z)) = -\alpha q' + \alpha q' = 0 = -\alpha q + \alpha q = Q_{a,q}(z, z) + Q_{a,q}(z, t_{a1}(z)) = \sum_{w\in\Gamma} Q_{a,q}(z, w)$.

Hence, in every case, we have $\sum_{w\in\Gamma} Q_{a,q'}(z, w) \leq \sum_{w\in\Gamma} Q_{a,q}(z, w)$; the lemma is thus proved. □

5.1. Extensions

Extending the result of this section to a general network is not obvious. There are some properties that can be shown easily. For instance, the st-monotonicity of the events corresponding to the external arrivals of data packets can be shown using the same arguments as above. However, there is a difference when data packets can be routed from one block to the other. In that case, the routing events are not monotone for the partial order obtained as the product order of partial orders on single blocks as defined in this section.

For example, consider the EPN illustrated in Figure 2 and the event $s_{i,j}$ that routes one packet from station i to station j. We have

$$t_{s_{i,j}}(x,y) = ((x - e_i + e_j)1_{[x_i>0]}, y - e_i)1_{[y_i>0]} + (x,y)1_{[y=0]}.$$

Now consider two states $z' = (x', y')$ and $z'' = (x'', y'')$ such that $x' = (1,1,1), y' = (1,0,0)$ and $x' = (1,1,1), y' = (0,0,0)$. Then $z' \preceq z''$, however,

$$t_{s_{1,2}}(z') = (0,2,1) \npreceq (1,1,1) = t_{s_{1,2}}(z'').$$

In the Appendix B, we define a partial order and an event representation that allows us to compare CTMCs representing tree-shaped EPN by comparing their JOB probability for each queue. Extending this result to a general network seems a difficult task, even when the network remains acyclic, but where there is a routing choice after a service of a packet. For example, in the EPN from Figure 2, the packets that are served by station 1 can be routed either to station 2 or station 3.

6. Conclusion

We study the EPN model with N blocks in which the energy queues have a finite capacity, and the leakage rate and the rate at which an energy packet arrives to the data queue of the same block depend on the number of energy packets. We also consider that the energy packets start the transfer. If an energy packet arrives to block i when the energy queue is full, jump-over blocking occurs, which means that the energy packets is sent to the data queue with probability q_i and it is lost with probability $1 - q_i$.

We generalize the model of [2] by consider a model with multiple classes of data packets. Besides, we consider that a data packet of class l is selected when an energy packet arrives can be according to one of the following disciplines: FCFS, PS or LCFS.

We show that there exists a vector of jump-over blocking probabilities $q = (q_1, \dots, q_n)$ such that the steady state probability distribution of packets in the queues has a product-form expression. The cases that are covered by this result include that the energy queue of block i is a M/M/1/B_i queue and a M/M/B_i/B_i queue.

For a monoclass directed tree network, we show that there exists a stochastic ordering according q. As a consequence, if we start two systems with different q

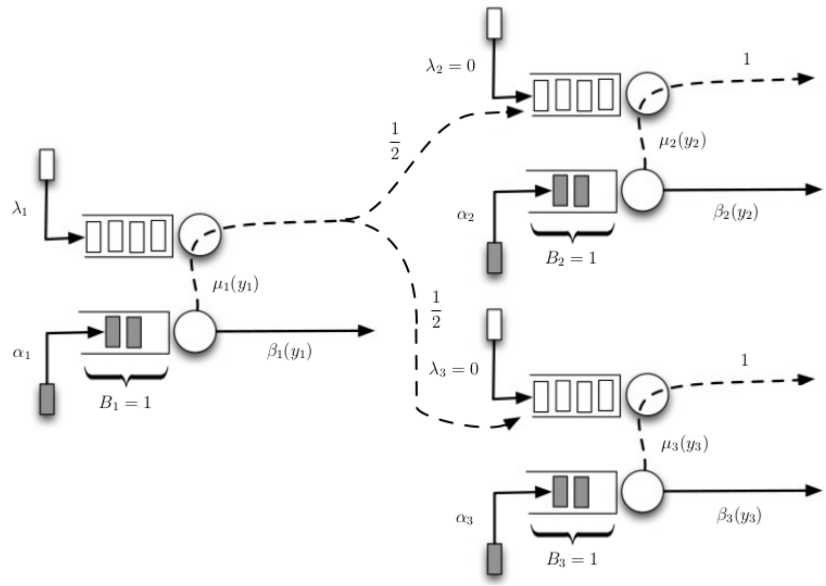

Figure 2: Three-blocks EPN.

vectors from the same initial state, then the cumulated number of data packets at any time instant t (and at the steady-state) in any subnetwork is stochastically smaller in the *st*-sense for the system with higher value of q. At the same time, the number of energy packets at any time instant t (and in steady state) is bigger in the *st*-sense for the higher value (in the sense of the product order) of q. Note that this stochastic ordering does not use the product form property of the steady state distributions so it allows comparison of systems with values of q for which we do not have a product form result. In particular, the system in which all energy packets are lost when the energy queue is full is upper bounded by the system with q vector corresponding to the product form of Section 4.

For future work, we are interested in extending the result of Theorem 1 to an EPN model where the data queues have also finite capacity. Another possibility for future research is to consider an EPN model where the data packets start the transfer. Moreover, the generalization of our results to phase-type service for the PS discipline is easy since each phase can be represented as a different class. However, we have seen that, for FCFS and preemptive LCFS, this generalization is not as straightforward as for PS and we would like to analyze it as well. Finally, we would also like to investigate whether the stochastic ordering result can be generalized to a network which is not a directed tree.

References

[1] E. Gelenbe, Energy packet networks: smart electricity storage to meet surges in demand, in: Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques, 2012, pp. 1–7.

[2] S. Samain, J. Doncel, A. Busic, J.-M. Fourneau, Energy packet networks with finite capacity energy queues, in: Proceedings of the 13th EAI International Conference on Performance Evaluation Methodologies and Tools, 2020, pp. 142–149.

[3] E. Gelenbe, Energy packet networks: Ict based energy allocation and storage, GreeNets 2011 (2011).

[4] E. Gelenbe, A sensor node with energy harvesting., SIGMETRICS Performance Evaluation Review 42 (2) (2014) 37–39.

[5] E. Gelenbe, A. Marin, Interconnected wireless sensors with energy harvesting, 22nd international conference, ASMTA 2015, Albena, Bulgaria. Proceedings; 2015. p. 87–99 (2016).

[6] E. Gelenbe, O. H. Abdelrahman, An energy packet network for mobile networks with energy harvesting, Nonlinear Theory and Its Applications, IEICE, vol. 9, no. 3, pp. 322–336 (2018).

[7] E. Gelenbe, E. Tugce Ceran, Central or distributed energy storage for processors with energy harvesting, 2015 Sustainable Internet and ICT for Sustainability (2015).

[8] J. Doncel, J.-M. Fourneau, Balancing energy consumption and losses with energy packet network models, in: 2019 IEEE International Conference on Fog Computing (ICFC), 2019, pp. 59–68. doi:10.1109/ICFC.2019.00017.

[9] J.-M. Fourneau, Modeling green data-centers and jobs balancing with energy packet networks and interrupted poisson energy arrivals, SN Computer Science 1 (1) (2019) 28. doi:10.1007/s42979-019-0029-5.
URL `https://doi.org/10.1007/s42979-019-0029-5`

[10] E. Gelenbe, E. Tugce Ceran, Energy packet network with energy harvesting, IEEE Access (2016).

[11] E. Gelenbe, Y. Zhang, Performance optimization with energy packets, IEEE Systems Journal 13 (4) (2019) 3770–3780.

[12] E. Gelenbe, Product-form queueing networks with negative and positive customers, Journal of applied probability 28 (3) (1991) 656–663.

[13] E. Gelenbe, G-networks with instantaneous customer movement., J. App. Probab. (1993).

[14] E. Gelenbe, G-networks with signals and batch removal, Probability in the Engineering and Informational Sciences 7 (3) (1993) 335–342.

[15] Y. A. El Mahjoub, H. Castel, J.-M. Fourneau, Energy packet networks with general service time distribution, IEEE Mascots 2020.

[16] O. H. Abdelrahman, E. Gelenbe, A diffusion model for energy harvesting sensor nodes, in: 2016 IEEE 24th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), IEEE, 2016, pp. 154–158.

[17] P. P. Ray, Energy packet networks: an annotated bibliography, SN Computer Science 1 (1) (2020) 6.

[18] Y. M. Kadioglu, E. Gelenbe, Product-form solution for cascade networks with intermittent energy, IEEE Systems Journal 13 (1) (2018) 918–927.

[19] Y. M. Kadioglu, Finite capacity energy packet networks, Probability in the Engineering and Informational Sciences 31 (4) (2017) 477–504. doi:10.1017/S0269964817000080.

[20] J. Doncel, J.-M. Fourneau, Energy packet networks with multiple energy packet requirements, Probability in the Engineering and Informational Sciences (2019) 1–19.

[21] N. M. Van Dijk, On jackson's product form with'jump-over'blocking (1988).

[22] W. A. Massey, An operator-analytic approach to the jackson network, Journal of Applied Probability 21 (2) (1984) 379–393. doi:10.2307/3213647.

[23] S. Balsamo, Properties and analysis of queueing network models with finite capacities, in: Performance Evaluation of Computer and Communication Systems, Springer, 1993, pp. 21–52.

[24] W. A. Massey, Stochastic orderings for markov processes on partially ordered spaces, Mathematics of operations research (2) (1987) 350–367.

[25] A. Muller, D. Stoyan, Comparison Methods for Stochastic Models and Risks, Wiley, New York, NY, 2002.

Appendix A. Proof of Lemmas of Section 4

Proof of Lemma 1. We distinguish three cases for the policy of the servers:

- PS case: in this case, we have:

$$\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} M_i^{(l)}(x_i \oplus e^{(l)}) = \frac{(|x_i|+1)\rho_i^{(l)}}{x_i^{(l)}+1} \frac{x_i^{(l)}+1}{|x_i|+1} = \rho_i^{(l)}$$

- FCFS case : in this case, we have:

$$\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} M_i^{(l)}(x_i \oplus e^{(l)}) = \rho_i^{(l)} \times 1 = \rho_i^{(l)}$$

- LCFS case : in this case, we have:

$$\frac{g_i(x_i \oplus e^{(l)})}{g_i(x_i)} M_i^{(l)}(x_i \oplus e^{(l)}) = \rho_i^{(l)} \times 1 = \rho_i^{(l)}$$

□

Proof of Lemma 2. We distinguish three cases for the policy of the servers:

- PS case: in this case, we have:

$$\frac{g_i(x_i \ominus e^{(l)})}{g_i(x_i)} E_i^{(l)}(x_i \ominus e^{(l)}) = \frac{x_i^{(l)}}{|x_i| \rho_i^{(l)}} 1_{[|x_i|>0]} = \frac{1}{\rho_i^{(l)}} \frac{x_i^{(l)}}{|x_i|} 1_{[|x_i|>0]} = \frac{1}{\rho_i^{(l)}} L_i^{(l)}(x_i \ominus e^{(l)})$$

- FCFS case : in this case, we have:

$$\frac{g_i(x_i \ominus e^{(l)})}{g_i(x_i)} E_i^{(l)}(x_i \ominus e^{(l)}) = \frac{1}{\rho_i^{(l)}} 1_{\left[r_{i,|x_i|}=l\right]} = \frac{1}{\rho_i^{(l)}} L_i^{(l)}(x_i \ominus e^{(l)})$$

- LCFS case : in this case, we have:

$$\frac{g_i(x_i \ominus e^{(l)})}{g_i(x_i)} E_i^{(l)}(x_i \ominus e^{(l)}) = \frac{1}{\rho_i^{(l)}} 1_{[r_{i,1}=l]} = \frac{1}{\rho_i^{(l)}} L_i^{(l)}(x_i \ominus e^{(l)})$$

□

Proof of Lemma 3. We distinguish three cases for the policy of the servers:

- PS case: in this case, we have:

$$1_{[|x_i|>0]} = \sum_{l=1}^{K} \frac{x_i^{(l)}}{|x_i|} 1_{[|x_i|>0]} = \sum_{l=1}^{K} \frac{x_i^{(l)}}{|x_i|} 1_{\left[x_i^{(l)}>0\right]} = \sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)})$$

- FCFS case : in this case, we have:

$$1_{[|x_i|>0]} = \sum_{l=1}^{K} 1_{\left[r_{i,|x_i|}=l\right]} = \sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)})$$

- LCFS case : in this case, we have:

$$1_{[|x_i|>0]} = \sum_{l=1}^{K} 1_{[r_{i,1}=l]} = \sum_{l=1}^{K} L_i^{(l)}(x_i \ominus e^{(l)})$$

$\square$

Proof of Lemma 4. Remember that $\gamma_i(y_i+1)\mu_i(y_i+1) = \alpha_i q_i$. We use Lemmas 1, 2 and the simple relations on the spate space $1_{[y_i<B_i]} + 1_{[y_i=B_i]} = 1$.

$$
\begin{aligned}
&\sum_{j=1}^{N}\sum_{l=1}^{K} \mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j(x_j\ominus e^{(l)})\\
&+\sum_{j=1}^{N}\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})E_j(x_j\ominus e^{(l)})\\
&=\sum_{j=1}^{N}\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}\left(1_{[y_i<B_i]}+1_{[y_i=B_i]}\right)M_i^{(l)}(x_i\oplus e^{(l)})E_j(x_j\ominus e^{(l)})\\
&=\sum_{j=1}^{N}\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}\frac{g_j(x_j\ominus e^{(l)})}{g_j(x_j)}p_{i,j}^{(l)}M_i^{(l)}(x_i\oplus e^{(l)})E_j(x_j\ominus e^{(l)})\\
&=\sum_{j=1}^{N}\sum_{l=1}^{K}\frac{\alpha_i q_i\rho_i^{(l)}}{\rho_j^{(l)}}p_{i,j}^{(l)}L_j(x_j\ominus e^{(l)})
\end{aligned}
$$

Using the same arguments, we get:

$$
\begin{aligned}
&\sum_{l=1}^{K}\mu_i(y_i+1)\gamma_i(y_i+1)\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}1_{[y_i<B_i]}M_i^{(l)}(x_i\oplus e^{(l)})\\
&+\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}1_{[y_i=B_i]}M_i^{(l)}(x_i\oplus e^{(l)})\\
&=\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}(1_{[y_i<B_i]}+1_{[y_i=B_i]})M_i^{(l)}(x_i\oplus e^{(l)})\\
&=\sum_{l=1}^{K}\alpha_i q_i\frac{g_i(x_i\oplus e^{(l)})}{g_i(x_i)}p_{i,s}^{(l)}M_i^{(l)}(x_i\oplus e^{(l)})\\
&=\sum_{l=1}^{K}\alpha_i q_i\rho_i^{(l)}p_{i,s}^{(l)}
\end{aligned}
$$

$\square$

Proof of Lemma 5.

$$\sum_{i=1}^{N}\sum_{l=1}^{K}\alpha_i q_i \rho_i^{(l)} p_{i,s}^{(l)} = \sum_{l=1}^{K}\sum_{i=1}^{N}\alpha_i q_i \rho_i^{(l)} - \sum_{l=1}^{K}\sum_{i=1}^{N}\alpha_i q_i \rho_i^{(l)} \sum_{j=1}^{N} p_{i,j}^{(l)}$$
$$= \sum_{l=1}^{K}\sum_{i=1}^{N}\alpha_i q_i \rho_i^{(l)} - \sum_{l=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i q_i \rho_i^{(l)} p_{i,j}^{(l)}$$
$$= \sum_{l=1}^{K}\sum_{i=1}^{N}\alpha_i q_i \rho_i^{(l)} - \sum_{l=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_j q_j \rho_j^{(l)} p_{j,i}^{(l)}$$

From the flow equation of the data packets, it follows that $\alpha_i q_i \rho_i^{(l)} = \lambda_i^{(l)} + \sum_{j=1}^{N}\alpha_j p_j \rho_j^{(l)} p_{j,i}^{l}$ and, as a result, the last term of the above expression is equal to

$$\sum_{l=1}^{K}\sum_{i=1}^{N}\left(\lambda_i^{(l)} + \sum_{j=1}^{N}\alpha_j q_j \rho_j^{(l)} p_{j,i}^{(l)}\right) - \sum_{l=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_j q_j \rho_j^{(l)} p_{j,i}^{(l)}$$
$$=\sum_{l=1}^{K}\sum_{i=1}^{N}\lambda_i^{(l)}.$$

And the desired result follows. □

Proof of Lemma 6. Let $\mathcal{W}([\![1,K]\!])$ the set of finite length words whose letters are in $[\![1,K]\!]$, and S_i the set in which evolves the state x_i of data queue i, that is

$$S_i = \begin{cases} \mathcal{W}([\![1,K]\!]) \text{ in the FCFS and LCFS case} \\ \mathbb{N}^K \text{ in the PS case} \end{cases}$$

Then we have:

$$\sum_{x\in\prod_{i=1}^{N} S_i}\ \sum_{y\in\prod_{i=1}^{N}[\![0,B_i]\!]} \pi(x,y)$$
$$= \sum_{x\in\prod_{i=1}^{N} S_i}\ \sum_{y\in\prod_{i=1}^{N}[\![0,B_i]\!]} \left(\prod_{i=1}^{N} C_i \left(1-\sum_{l=1}^{K}\rho_i^{(l)}\right) g_i(x_i) \prod_{j=1}^{y_i}\gamma_i(j)\right)$$
$$= \prod_{i=1}^{N}\left(\sum_{x_i\in S_i}\left(1-\sum_{l=1}^{K}\rho_i^{(l)}\right) g_i(x_i)\right)\left(\sum_{y_i=0}^{B_i}\left(C_i\prod_{j=1}^{y_i}\gamma_i(j)\right)\right)$$
$$= \left(\prod_{i=1}^{N}\left(\sum_{x_i\in S_i}\left(1-\sum_{l=1}^{K}\rho_i^{(l)}\right) g_i(x_i)\right)\right)\left(\prod_{i=1}^{N}\left(\sum_{y_i=0}^{B}\left(C_i\prod_{j=1}^{y_i}\gamma_i(j)\right)\right)\right)$$

We prove that both of the left term and the right term of this product are equal to 1.

We first prove that $\sum_{x_i \in S_i} g(x_i) = 1$ in the FCFS, LCFS and PS case:

- LCFS or FCFS case: in this case, we have $S_i = \mathcal{W}(\llbracket 1, K \rrbracket)$, and

$$\sum_{x_i \in S_i} g(x_i) = \sum_{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket)} \prod_{k=1}^{|x_i|} \rho_i^{(r_{i,k})} = \sum_{j=0}^{+\infty} \sum_{\substack{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket) \\ |x_i|=j}} \prod_{k=1}^{|x_i|} \rho_i^{(r_{i,k})}$$

$$= \sum_{j=0}^{+\infty} \sum_{\substack{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket) \\ |x_i|=j}} \prod_{k=1}^{|x_i|} \rho_i^{(r_{i,k})}$$

  Now, if we note $x_i^{(l)}$ the number of data packets of class l in a data queue whose state is x_i, then for any $j \in \llbracket 1, N \rrbracket$, by summation we get:

$$\sum_{\substack{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket) \\ |x_i|=j}} \prod_{k=1}^{|x_i|} \rho_i^{(r_{i,k})} = \sum_{\substack{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket) \\ |x_i|=j}} \prod_{l=1}^{K} \left(\rho_i^{(l)}\right)^{x_i^{(l)}}$$

$$= \sum_{\substack{s_1,\dots,s_K \\ \sum_{l=1}^K s_l = j}} \left(\begin{array}{c} j \\ s_1, \dots, s_K \end{array} \right) \prod_{l=1}^{K} \left(\rho_i^{(l)}\right)^{s_l}$$

$$= \left(\sum_{l=1}^{K} \rho_i^{(l)} \right)^j$$

  and hence, we have:

$$\sum_{j=0}^{+\infty} \sum_{\substack{x_i \in \mathcal{W}(\llbracket 1,K \rrbracket) \\ |x_i|=j}} \prod_{k=1}^{|x_i|} \rho_i^{(r_{i,k})} = \sum_{j=0}^{+\infty} \left(\sum_{l=1}^{K} \rho_i^{(l)} \right)^j = \frac{1}{1 - \sum_{l=1}^K \rho_i^{(l)}}$$

- PS case: in this case, we have $S_i = \mathbb{N}^K$, and

$$\sum_{x_i \in S_i} g(x_i) = \sum_{x_i \in \mathbb{N}^K} \left(\begin{array}{c} |x_i| \\ x_i(1), \dots, x_i^{(K)} \end{array} \right) \prod_{l=1}^{K} \left(\rho_i^{(l)}\right)^{x_i^{(l)}}$$

$$= \sum_{j=0}^{+\infty} \sum_{\substack{x_i \in \mathbb{N}^K \\ \sum_{l=1}^K x_i^{(l)} = j}} \left(\begin{array}{c} j \\ x_i^{(1)}, \dots, x_i^{(K)} \end{array} \right) \prod_{l=1}^{K} \left(\rho_i^{(l)}\right)^{x_i^{(l)}}$$

$$= \sum_{j=0}^{+\infty} \left(\sum_{l=1}^{K} \rho_i^{(l)} \right)^j = \frac{1}{1 - \sum_{l=1}^K \rho_i^{(l)}}$$

Hence, for the left term, we have:

$$\prod_{i=1}^{N}\left(\sum_{x_i\in S_i}\left(1-\sum_{l=1}^{K}\rho_i^{(l)}\right)g_i(x_i)\right)=\prod_{i=1}^{N}\left(\left(1-\sum_{l=1}^{K}\rho_i^{(l)}\right)\sum_{x_i\in S_i}g_i(x_i)\right)$$
$$=\prod_{i=1}^{N}\left(\frac{1-\sum_{l=1}^{K}\rho_i^{(l)}}{1-\sum_{l=1}^{K}\rho_i^{(l)}}\right)=\prod_{i=1}^{N}1=1$$

Now, for the right term, we have:

$$\prod_{i=1}^{N}\left(\sum_{y_i=0}^{B}C_i\left(\prod_{j=1}^{y_i}\gamma_i(j)\right)\right)$$
$$=\prod_{i=1}^{N}\left(C_i\left(\sum_{y_i=0}^{B_i}\prod_{j=1}^{y_i}\gamma_i(j)\right)\right)$$
$$=\prod_{i=1}^{N}\left(\frac{\sum_{y_i=0}^{B_i}\prod_{j=1}^{y_i}\gamma_i(j)}{\sum_{y_i=0}^{B_i}\prod_{j=1}^{y_i}\gamma_i(j)}\right)=1$$

and hence, the lemma is proved.

□

Appendix B. Background on Stochastic Ordering

We first recall what is strong stochastic order for a pair of random variables:

Definition 1. *Let $(S,\preceq_S)$ be a partially ordered space and X and Y two random variables on S. X is smaller than Y in a strong stochastic sense, noted $X\preceq_{st}Y$, if*

$$E[f(X)]\le E[f(Y)] \text{ for all increasing functions } f,$$

provided that the expectations exist.

Strong stochastic comparison of random variables on a partially ordered set can be characterized by means of increasing sets. A subset $\Gamma\subseteq S$ is called an increasing set if its indicator function 1_Γ is increasing. It follows that Γ is an increasing set if and only if $x\in\Gamma$ and $x\preceq_S y$ imply $y\in\Gamma$. The following characterization is often used as definition of st-order on a partially ordered space [24]. The proof can be found in [25].

Lemma 11. *$X\preceq_{st}Y$ if and only if $P(X\in\Gamma)\le P(Y\in\Gamma)$, for all increasing sets $\Gamma\subseteq S$.*

We now define strong stochastic order for two processes:

Definition 2. *Let* $(S, \preceq_S)$ *be a partially ordered space and* X *and* Y *two processes on* S *indexed by* $\mathbb{R}_+$. X *is smaller than* Y *in a strong stochastic sense, noted* $X \preceq_{st} Y$ *iff* $\forall t \in \mathbb{R}_+, X_t \preceq_{st} Y_t$.

We know that for CTMCs, the following characterization of strong stochastic order holds:

Theorem 4. *[24, Thm 5.3] Let* $X = \{X_t\}_{t\geq 0}$ *and* $Y = \{Y_t\}_{t\geq 0}$ *be two CTMC with infinitesimal generators* Q *and* R. *Then* $X_t \preceq_{st} Y_t$, $\forall t \geq 0$ *if and only if:*

- $X_0 \preceq_{st} Y_0$,
- *for all* $u, v \in S$ *such that* $u \preceq_S v$ *and for all increasing sets* $\Gamma \subseteq S$ *such that* $u \in \Gamma$ *or* $v \notin \Gamma$ *we have:*

$$\sum_{w\in\Gamma} Q(u,w) \leq \sum_{w\in\Gamma} R(v,w).$$

However, the conditions stated in the previous theorem may be difficult to check. Hence, we present a sufficient condition in corollary 3, which is easier to verify:

Definition 3. *A CTMC* $X = \{X_t\}_{t\geq 0}$ *is monotone if for any two initial distributions* μ *and* ν *of* X_0 *such that* $\mu \preceq_{st} \nu$ *we have:*

$$\forall t > 0, X_t^{\mu} \preceq_{st} X_t^{\nu},$$

where X_t^{μ} *denotes that the initial distribution of* X_0 *is* μ.

Theorem 5. *[24, Thm 5.2] A CTMC* $X = \{X_t\}_{t\geq 0}$ *with an infinitesimal generator* Q *is st-monotone if and only if for all* $u, v \in S$ *such that* $u \preceq_S v$ *and for all increasing sets* $\Gamma \subseteq S$ *such that* $u \in \Gamma$ *or* $v \notin \Gamma$:

$$\sum_{w\in\Gamma} Q(u,w) \leq \sum_{z\in\Gamma} Q(v,w).$$

Definition 4. *Let* Q *and* R *be two generators. Then* $Q \preceq_{st} R$ *if for any* $u \in S$ *and for all increasing sets* $\Gamma \subseteq S$ *we have:*

$$\sum_{w\in\Gamma} Q(u,w) \leq \sum_{w\in\Gamma} R(u,w).$$

We have the following sufficient condition:

Corollary 3. *Let* $X = \{X_t\}_{t\geq 0}$ *and* $Y = \{Y_t\}_{t\geq 0}$ *be two CTMC with infinitesimal generators* Q *and* R. *Then* $X_t \preceq_{st} Y_t$, $\forall t \geq 0$ *if*

- $X_0 \preceq_{st} Y_0$,
- *there is an st-monotone generator* A *such that*

$$Q \preceq_{st} A \preceq_{st} R.$$

Appendix C. Stochastic comparison for directed trees

In this section, we extend the results of section 5 on the single block model, in the case when the underlying routing graph of the queueing network is a directed tree. This routing topology can be of particular interests for sensor networks where the nodes are powered by solar panels and can send the collected information only to the nearest neighbor on their path to the central node (root).

We start by recalling the definition of a directed tree:

Definition 5. *A directed tree is a finite simply connected directed graph* $T = (V, E)$ *such that there is a node* r*, called the root, such that* r *has only incoming arcs, and such that any other node has exactly one outgoing arc.*

A vertex which has no incoming arc is called a leaf.

A vertex u *is called a child of some vertex* v*, noted* $u \prec v$*, if there exists an arc from* u *to* v*. A vertex* v *is called the parent of some vertex* u *if we have* $u \prec v$*, that is,* u *is a child of* v*. The parent of* u *is noted* $p(u)$*.*

A vertex u *is a successor of some vertex* v *iff there exists a directed path from* u *to* v *in* T*. Denote by* $\Gamma(v)$ *the set containing* v *and all of its successors. Hence,* $(\Gamma(v), E \cap (\Gamma(v) \times \Gamma(v)))$ *is the* subtree rooted at vertex v.

An example of directed tree is given in the figure below. Its root is a. Its set of leaves is $\{d, e, g, h\}$. The set of children of a is $\{b, c\}$. The successor set of c is $\Gamma(c) = \{c, f, g, h\}$. The subtree rooted at c is $(\{c, f, g, h\}, \{(g, f), (h, f), (f, c)\})$.

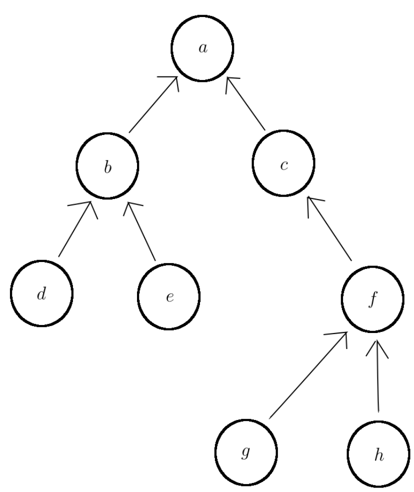

Figure C.3: An example of a directed tree

In this context, we study the influence of the probability that an energy packet is sent to the data queue when the jump-over blocking occurs (i.e., when the energy queue is full). This is be done by defining an order which compares processes representing the same directed tree EPN, but with possibly different JOB probabilities at their batteries.

We define a partial order of interest for a directed tree EPN:

Definition 6 (Subtree order)**.** *Given a directed tree EPN with state space S, for two states $z' = (x', y')$ and $z'' = (x'', y'')$ in S, we say that $z' \preceq_{tree} z''$ if*

$$\forall i \in [\![1, n]\!], \sum_{j \in \Gamma(i)} x'_j \leq \sum_{j \in \Gamma(i)} x''_j$$

$$\forall i \in [\![1, n]\!], y'_i \geq y''_i$$

We call that order the subtree order over S.

Intuitively, it is preferable to have, for each node i:

- the least amount of data packet in the queues of the subtree network rooted at i
- the greatest amount of energy packets in the battery of node i

In the following, $\preceq_{st}$ will be the strong stochastic order associated to the partial order $\preceq_{tree}$.

In the spirit of Section 5, we wish to build an event representation which will allow us the use of Massey's theorem; for a directed tree EPN, we use the following sets of events:

- $\mathcal{E}_{a^1} = \{a_i^1 | i \in [\![1, n]\!]\}$
- $\mathcal{E}_{a^2} = \{a_i^2 | i \in [\![1, n]\!]\}$
- $\mathcal{E}_a = \mathcal{E}_{a^1} \cup \mathcal{E}_{a^2}$
- $\mathcal{E}_d = \{d_i | i \in [\![1, n]\!]\}$
- $\mathcal{E}_b = \{b_i | i \in [\![1, n]\!]\}$
- $\mathcal{E}_s = \{s_i | i \in [\![1, n]\!]\}$
- $\mathcal{E}_{EPN} = \mathcal{E}_a \cup \mathcal{E}_d \cup \mathcal{E}_b \cup \mathcal{E}_s$

where for each block i, we have:

- a_i^1: *arrival of an energy packet of type 1 (the arriving energy packet serves a data packet when the battery is full and is queued in the battery otherwise)* with rate $\tau_{a_i^1}(z) = q_i \alpha_i$;
- a_i^2: *arrival of an energy packet of type 2 (the arriving data packet is leaked when the battery is full and queued in the battery otherwise)* with rate $\tau_{a_i^2}(z) = (1 - q_i)\alpha_i$;

- d_i: *arrival of a data packet* with rate $\tau_{d_i}(z) = \lambda_i$;
- b_i: *leakage of an energy packet* with rate $\tau_{b_i}(z) = \beta_i(y_i)$;
- s_i: *service of a data packet, triggered by an energy packet* with rate $\tau_{s_i}(z) = \mu_i(y_i)$.

The reason for splitting arrivals of energy packets into two types of events (type 1 and type 2) is purely to be able to describe the effect of the event by deterministic functions. To each event e, we associate a transition function $t_e : S \to S$ defined as follows:

- a_i^1: $t_{a_i^1}(x, y) = (x, y + e_i)1_{[y_i<B_i]} + (x + (e_{p(i)}1_{[i\neq r]} - e_i)1_{[x_i>0]}, y)1_{[y_i=B_i]}$;
- a_i^2: $t_{a_i^2}(x, y) = (x, y + e_i)1_{[y_i<B_i]} + (x, y)1_{[y_i=B_i]}$;
- d_i: $t_{d_i}(x, y) = (x + e_i, y)$;
- b_i: $t_{b_i}(x, y) = (x, (y - e_i)^+)$;
- s_i: $t_{s_i}(x, y) = (x + (e_{p(i)}1_{[i\neq r]} - e_i)1_{[x_i>0]}, y - e_i)1_{[y_i>0]} + (x, y)1_{[y_i=0]}$,

and a generator $Q_e = \Delta(\tau_e)(E(t_e) - I)$, with $\Delta(\tau_e)$ the diagonal matrix of rates and $E(t_e) = (1_{[t(z')=z'']})_{(z',z'')\in S\times S}$. Let $q_i \in [0, 1]$ and $Q_{a_i,q_i} = Q_{a_i^1} + Q_{a_i^2}$ where the rate function for the events a_i^1 and a_i^2 are respectively $\alpha_i q_i$ and $\alpha_i(1 - q_i)$. The generator of the CTMC associated to the directed tree EPN with JOB probability vector q is then given by

$$Q_q = \sum_i (Q_{d_i} + Q_{a_i,q_i} + Q_{b_i} + Q_{s_i}) .$$

In the two following lemmas, we show that all the events $e \in \mathcal{E}_{EPN}$ are st-monotone for the cases where energy queues are M/M/1/B queues or M/M/B/B queues. We study first the M/M/1/B case:

Lemma 12. *Let $e \in \mathcal{E}_{EPN}$ in a EPN whose routing graph is a directed tree, when the corresponding energy queue (i.e. the energy queue affected by the event e) is an M/M/1/B queue. Then, e is st-monotone.*

Proof. All the rates are state-independent, i.e., $\tau_e(z)$ is a constant that only depends on $e \in \mathcal{E}_{EPN}$. Therefore, conditions 2) and 3) of Theorem 3 are never verified and, as a consequence, to show that an event e is st-monotone, it is enough to show that condition 1) a) of Theorem 3 are satisfied.

To show this condition, for each event e, we partition the state space S into "types" of states such that t_e is a translation with respect to $\preceq_{tree}$ on each "type set" (a subset of states of one given type), that is, if (a) is a type of states for e and $A_{(a)}$ is the subset of states of type (a), then $\forall z \in A_{(a)}, t_e(z) = z + v_{(a)}$, with $v_{(a)} = (v^x_{(a)}, v^y_{(a)})$ a vector which depends only on the type (a); hence, on each of type sets, condition 1) a) will hold; moreover, when considering couples $(z', z'') \in S^2$ such that $z' \preceq z''$ and z' is not of the same type as z'', some cases will be not be possible; for instance, if a type (a) requires $z = (x, y)$ to be such that $y_i = B_i$ and a type (b) requires $y_i < B_i$, then we can have z' of type (a) and z'' of type (b), but not the converse; and if a type (a) requires $z = (x, y)$ to be such that $y_i > 0$ and a type (b) requires $y_i = 0$, then we can have z' of type (a) and z'' of type (b), but not the converse.

In the following, it will be assumed that the states $z = (x, y), z' = (x', y')$ and $z'' = (x'', y'')$. For any set of nodes $A \subseteq [\![1, N]\!]$, and any $z \in S$, we note $f_A(z) = \sum_{i \in A} x_i$. Hence, if $t_e(z) = z + v_{(a)}$ for z of type (a), and $v_{(a)} = (v^x_{(a)}, v^y_{(a)})$, then we have $f_A(t_e(z)) = f_A(z) + \sum_{i \in A} (v^x_{(a)})_i$.

In the following, we note r the root of the routing graph. Hence, r is the only node which does not have any parent.

We now examine each type of event and show that they are all st-monotone.

Event d_i is st-monotone for all i. If $z' \preceq_{tree} z''$, we have $t_{d_i}(z') = (x' + e_i, y') \preceq_{tree} (x'' + e_i, y'') = t_{d_i}(z'')$. Hence, condition 1) a) holds.

Event b_i is st-monotone for all i. For $z \in S$, we distinguish two types:

(i) $y_i > 0$,

(ii) $y_i = 0$.

Let $(z', z'') \in S^2$ such that $z' \preceq_{tree} z''$. Transition t_{b_i} is a translation by $(0, -e_i)$ on $A_{(i)}$, and by $(0, 0)$ on $A_{(ii)}$ (i.e. $t_{b_i}(x, y) = (x, y - e_i)$, for $(x, y) \in A_{(i)}$, and $t_{b_i}(x, y) = (x, y)$, for $(x, y) \in A_{(ii)}$). Thus, if z' and z'' have same type then condition 1) a) holds. If z' and z'' are of types (i) and (ii), then z' is of type (i) and z'' is of type (ii) (the converse is not possible due to the definition of partial order $\preceq_{tree}$), and $t_{b_i}(z') = (x', y' - e_i) \preceq_{tree} (x'', 0) = t_{b_i}(z'')$. Hence, condition 1) a) holds.

Event s_i is st-monotone all i. For $z \in S$, we distinguish three types:

(i) $x_i > 0, y_i > 0$

(ii) $x_i = 0, y_i > 0$

(iii) $y_i = 0$

Let $(z', z'') \in S^2$ such that $z' \preceq_{tree} z''$. We distinguish five cases:

- *z' and z'' have the same type:* as t_{s_i} is a translation by $(-e_i + e_{p(i)} 1_{[r \neq i]}, -e_i)$ on $A_{(i)}$, by $(0, -e_i)$ on $A_{(ii)}$ and by $(0, 0)$ on $A_{(iii)}$, condition 1) a) holds.
- *z' is of type (i) and z'' is of type (ii):* we have $t_{s_i}(z') = (x' - e_i + e_{p(i)} 1_{[r \neq i]}, y' - e_i)$ and $t_{s_i}(z'') = (x'', y'' - e_i)$. Remark that if $i \neq r$ and $p(i) \in \Gamma(k)$, then also $i \in \Gamma(k)$. So we need to consider the following cases:
  - For all k such that $i \notin \Gamma(k)$,
  $$f_{\Gamma(k)}(t_{s_i}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z'')).$$
  - For $k = i$,
  $$f_{\Gamma(i)}(t_{s_i}(z')) = f_{\Gamma(i)}(z') - 1 \leq f_{\Gamma(i)}(z') \leq f_{\Gamma(j)}(z'') = f_{\Gamma(i)}(t_{s_i}(z'')).$$

- If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$ (and thus also $i \in \Gamma(k)$),

$$\begin{aligned} f_{\Gamma(k)}(t_{s_i}(z')) &= \sum_{j \in \Gamma(k)\setminus\{i,p(i)\}} x'_j + (x'_i - 1) + (x'_{p(i)} + 1) \\ &= f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z'')). \end{aligned}$$

- $\forall k \in [\![1, n]\!], y'_k - 1_{[k=i]} \geq y''_k - 1_{[k=i]}$.

Hence, condition 1) a) holds.

- *z' is of type (ii) and z'' is of type (i):* we have $t_{s_i}(z') = (x', y' - e_i)$ and $t_{s_i}(z'') = (x'' - e_i + e_{p(i)} 1_{[r \neq i]}, y'' - e_i)$. Then:
  - For all k such that $i \notin \Gamma(k)$, $f_{\Gamma(k)}(t_{s_i}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z''))$.
  - For $k = i$, $f_{\Gamma(i)}(t_{s_i}(z')) = f_{\Gamma(i)}(z') = \sum_{j \prec i} f_{\Gamma(j)}(z') + 0 \leq \sum_{j \prec i} f_{\Gamma(j)}(z'') + (x''_i - 1) = f_{\Gamma(i)}(t_{s_i}(z''))$.
  - If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$, $f_{\Gamma(k)}(t_{s_i}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = \sum_{j \in \Gamma(k)\setminus\{i,p(i)\}} x''_j + (x''_i - 1) + (x''_{p(i)} + 1) = f_{\Gamma(k)}(t_{s_i}(z''))$.
  - $\forall k \in [\![1, n]\!], y'_k - 1_{[k=i]} \geq y''_k - 1_{[k=i]}$.

  Hence, condition 1) a) holds.

- *z' is of type (i), z'' is of type (iii)* (remark that converse is not possible): then $t_s(z') = (x' - e_i + e_{p(i)} 1_{[r \neq i]}, y' - e_i)$ and $t_s(z'') = (x'', y'')$. We have:
  - For all k such that $i \notin \Gamma(k)$, $f_{\Gamma(k)}(t_{s_i}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z''))$.
  - For $k = i$, $f_{\Gamma(i)}(t_{s_i}(z')) = f_{\Gamma(i)}(z') - 1 \leq f_{\Gamma(i)}(z'') - 1 \leq f_{\Gamma(i)}(z'') = f_{\Gamma(i)}(t_{s_i}(z''))$.
  - If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$,

$$\begin{aligned} f_{\Gamma(k)}(t_{s_i}(z')) &= \sum_{j \in \Gamma(k)\setminus\{i,p(i)\}} x'_j + (x'_i - 1) + (x'_{p(i)} + 1) = f_{\Gamma(k)}(z') \\ &\leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z'')). \end{aligned}$$

  - $\forall k \in [\![1, n]\!], y'_k - 1_{[k=i]} \geq y''_k$.

  Hence, condition 1) a) holds.

- *z' is of type (ii) and z'' is of type (iii)* (remark that converse is not possible): then $t_{s_i}(x') = (x', y' - e_i)$ and $t_{s_i}(z'') = (x'', y'')$. But we have:
  - $\forall k \in [\![1, n]\!], f_{\Gamma(k)}(t_{s_i}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{s_i}(z''))$.
  - $\forall k \in [\![1, n]\!], y'_k - 1_{[k=i]} \geq y''_k$.

  Hence, condition 1) a) holds.

Hence, we have $t_{s_i}(z') \preceq_{tree} t_{s_i}(z'')$.

Event a_i^2 is st-monotone for all i. For $z \in S$, we distinguish two types:

(i) $y_i < B_i$,

(ii) $y_i = B_i$.

Let $(z', z'') \in S^2$ such that $z' \preceq_{tree} z''$. As $t_{a_i^2}$ is a translation by $(0, e_i)$ on $A_{(i)}$ and by $(0,0)$ on $A_{(ii)}$, condition 1) a) holds when z' and z'' have the same type. If z' is of type (ii) and z'' is of type (i) (converse is not possible), then $t_{a_i^2}(z') = (x', y') \preceq_{tree} (x'', y'' + e_i) = t_{a_i^2}(z'')$. Hence, condition 1) a) holds.

Event a_i^1 is st-monotone for all i. For $z \in S$, we distinguish three types:

- (i) $y_i < B_i$,
- (ii) $x_i > 0, y_i = B_i$,
- (iii) $x_i = 0, y_i = B_i$.

Let $(z', z'') \in S^2$ such that $z' \preceq_{tree} z''$; we distinguish cases:

- *z' and z'' have the same type:* as $t_{a_i^1}$ is a translation by $(0, e_i)$ on $A_{(i)}$, by $(-e_i + e_{p(i)}1_{[r \neq i]}, -e_i)$ on $A_{(ii)}$ and by $(0,0)$ on $A_{(iii)}$, condition 1) a) holds.
- *z' is of type (ii), z'' is of type (i)* (converse is not possible): then $t_{a_i^1}(z') = (x' - e_i + e_{p(i)}1_{[r \neq i]}, y')$ and $t_{a_i^1}(z'') = (x'', y'' + e_i)$. We have:
  - For all k such that $i \notin \Gamma(k)$, $f_{\Gamma(k)}(t_{a_i^1}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z''))$.
  - For $k = i$, $f_{\Gamma(i)}(t_{a_i^1}(z')) = f_{\Gamma(i)}(z') - 1 \leq f_{\Gamma(i)}(z') \leq f_{\Gamma(i)}(z'') = f_{\Gamma(i)}(t_{a_i^1}(z''))$.
  - If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$,
  $$\begin{aligned} f_{\Gamma(k)}(t_{a_i^1}(z')) &= \sum_{j \in \Gamma(k) \setminus \{i, p(i)\}} x'_j + (x'_i - 1) + (x'_{p(i)} + 1) \\ &= f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z'')). \end{aligned}$$
  - $\forall k \in [\![1, n]\!] \setminus \{i\}, y'_k \geq y''_k$.
  - $B_i \geq y''_i + 1$.

  Hence, condition 1) a) holds.
- *z' is of type (iii), z'' is of type (i)* (converse is not possible): then $t_{a_i^1}(z') = (x', y')$ and $(x'', y'' + e_i) = t_{a_i^1}(z'')$. We have:
  - $\forall k \in [\![1, n]\!]$, $f_{\Gamma(k)}(t_{a_i^1}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z''))$.
  - $\forall k \in [\![1, n]\!] \setminus \{i\}, y'_k \geq y''_k$

- $B_i \geq y''_i + 1$

Hence, condition 1) a) holds.

- *z' is of type (ii), z'' is of type (iii):* then $t_{a_i^1}(z') = (x'' - e_i + e_{p(i)} 1_{[r \neq i]}, y')$ and $(x'', y'') = t_{a_i^1}(z'')$.
  - For all k such that $i \notin \Gamma(k)$, $f_{\Gamma(k)}(t_{a_i^1}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z''))$.
  - For $k = i$, $f_{\Gamma(i)}(t_{a_i^1}(z')) = f_{\Gamma(i)}(z') - 1 \leq f_{\Gamma(i)}(z') \leq f_{\Gamma(i)}(z'') = f_{\Gamma(i)}(t_{a_i^1}(z''))$
  - If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$,

$$\begin{aligned} f_{\Gamma(k)}(t_{a_i^1}(z')) &= \sum_{j \in \Gamma(k) \setminus \{i, p(i)\}} x'_j + (x'_i - 1) + (x'_{p(i)} + 1) \\ &= f_{\Gamma(k)}(z') + 1 - 1 = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z'')). \end{aligned}$$

  - $\forall k \in [\![1, n]\!], y'_k \geq y''_k$.

  Hence, condition 1) a) holds.

- *z' is of type (iii) and z'' is of type (ii):* then $t_{a_i^1}(z') = (x', y')$ and $(x'' - e_i + e_{p(i)} 1_{[r \neq i]}, y'') = t_{a_i^1}(z'')$. But we have:
  - For all k such that $i \notin \Gamma(k)$, $f_{\Gamma(k)}(t_{a_i^1}(z')) = f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(t_{a_i^1}(z''))$.
  - For $k = i$, $f_{\Gamma(i)}(t_{a_i^1}(z')) = f_{\Gamma(i)}(z') = f_{\Gamma(i)}(z'') - 1 = f_{\Gamma(i)}(t_{a_i^1}(z''))$.
  - If $i \neq r$, then for k such that $p(i) \in \Gamma(k)$,

$$\begin{aligned} f_{\Gamma(k)}(t_{a_i^1}(z')) &= f_{\Gamma(k)}(z') \leq f_{\Gamma(k)}(z'') = f_{\Gamma(k)}(z'') + 1 - 1 \\ &= \sum_{j \in \Gamma(k) \setminus \{i, p(i)\}} x''_j + (x''_i - 1) + (x''_{p(i)} + 1) = f_{\Gamma(k)}(t_{a_i^1}(z'')). \end{aligned}$$

  - $\forall k \in [\![1, n]\!], y'_k \geq y''_k$.

  Hence, condition 1) a) holds.

□

We now consider the M/M/B/B case for the energy queue.

Lemma 13. *Let $e \in \mathcal{E}_{EPN}$ in a EPN whose routing graph is a directed tree, when the corresponding energy queue (i.e. the energy queue affected by the event e) is an M/M/B/B queue. Then, e is st-monotone.*

Proof. The events a_i^1, a_i^2 and d_i, $i \in [\![1, n]\!]$ are exactly the same as in the M/M/1/B case, so the proof is identical as in the previous lemma.

The case of events b_i and s_i, $i \in [\![1, n]\!]$, is different, because we have $\beta_i(y_i) = y_i\beta_i$ and $\mu_i(y_i) = y_i\mu_i$; hence, in addition to condition 1)a), we will also verify condition 3) for these events. Remark that the proof of condition 1) a) remains exactly the same, as we only changed the rates of these events but not their transition functions.

Event b_i is st-monotone for any i. We consider $z' = (x', y')$ and $z'' = (x'', y'')$ where $z' \preceq_{tree} z''$.

- If $y_i' = y_i''$, then $\beta_i(y_i') = \beta_i(y_i'')$ and condition 1) a) is sufficient.
- If $y_i' > y_i''$, then $\beta_i(y_i') > \beta_i(y_i'')$ and therefore, we also need to verify that condition 3) of Theorem 3 is satisfied, i.e. that $t_{b_i}(z') \preceq_{tree} z''$. As $y_i' > y_i'' \geq 0$, we have $y_i' - 1 \geq y_i''$, thus $t_{b_i}(z') = (x', y' - e_i) \preceq_{tree} (x', y'') \preceq_{tree} z''$.

Event s_i is st-monotone for any i. This event occurs with rate $y_i\mu_i$. We consider $z' = (x', y')$ and $z'' = (x'', y'')$ where $z' \preceq_{tree} z''$. As before, we only need to consider the states such that $y_i' \neq y_i''$. Since $z' \preceq_{tree} z''$, we have $y_i' > y_i''$. To show 3) of Theorem 3, we divide our state space as follows:

- (i) states z for which $x_i > 0, y_i > 0$,
- (ii) states z for which $x_i = 0, y_i > 0$,
- (iii) states z for which $y_i = 0$

If $A_{(i)}, A_{(ii)}$ and $A_{(iii)}$ are the subsets of states of type (i), (ii) and (iii) respectively, then t_{s_i} is a translation by $(-e_i + e_{p(i)}1_{[r\neq i]}, -e_i)$ on $A_{(i)}$, by $(0, -e_i)$ on $A_{(ii)}$ and by $(0, 0)$ on $A_{(iii)}$.

In order to prove condition 3), we distinguish the following cases:

- *z' is of type (i):* We have $t_{s_i}(z') = (x' - e_i + e_{p(i)}1_{[r\neq i]}, y' - e_i) \preceq_{tree} (x' - e_i + e_{p(i)}1_{[r\neq i]}, y'')$, as $y_i' > y_i''$. By arguments similar to the ones used in the proof of the previous lemma, we have $(x' - e_i + e_{p(i)}1_{[r\neq i]}, y'') \preceq_{tree} (x'', y'') = z''$, and hence condition 3) holds.
- *z' is of type (ii):* as $y_i' > y_i''$, $t_{s_i}(z') = (x', y' - e_i) \preceq_{tree} (x'', y'') = z''$, and hence condition 3) holds.
- *z' is of type (iii):* in this case, from $z' \preceq_{tree} z''$ it follows that z'' is also of type (iii). Thus, the service rate is zero for both z' and z'', and we do not need to prove condition 3).

Hence, condition 3) holds for the event s_i.

□

By summing the generators of the events, we obtain that the generator Q_q of the CTMC associated with the directed tree EPN model with JOB probability vector q is st-monotone. We now want to show that when $(q, q') \in [0,1]^n \times [0,1]^n$ are such that $q \leq q'$ for the product order, then $Q_{q'} \preceq_{st} Q_q$. Again, our event representation is will help us, due to Lemma 9 of Section 5.

As $Q_e \preceq_{st} Q_e$ for any $e \in \mathcal{E}_b \cup \mathcal{E}_d \cup \mathcal{E}_s$ and Q_e is st-monotone for any $e \in \mathcal{E}_{EPN}$ when the energy queue is either M/M/1/B or M/M/B/B, we only need to show the following lemma:

Lemma 14. *For any $i \in [\![1, n]\!]$, if $q_i \leq q'_i$, then $Q_{a_i,q'_i} \preceq_{st} Q_{a_i,q_i}$.*

Proof. We choose $z = (x, y) \in S$, and we distinguish three cases:

- Case 1: $y_i < B_i$: in this case, $Q_{a_i,q_i}(z, z) = Q_{a_i,q'_i}(z, z) = -\alpha_i, Q_{a_i,q_i}(z, z+(0, e_i)) = Q_{a_i,p'}(z, z + (0, e_i)) = \alpha_i$, and for any $z' \in S \setminus \{z, z + (0, e_i)\}$, we have $Q_{a_i,q_i}(z, z') = Q_{a_i,q'_i}(z, z') = 0$. Hence, $Q_{a_i,q_i}(z, .) = Q_{a_i,q'_i}(z, .)$, and hence $\forall\Gamma$ increasing set, $\sum_{w\in\Gamma} Q_{a_i,q_i}(z, w) = \sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w)$.

- Case 2: $x_i > 0, y_i = B_i$: in this case, $Q_{a_i,q_i}(z, z) = -\alpha_i q_i$, $Q_{a_i,q'_i}(z, z) = -\alpha_i q'_i$, $Q_{a_i,q_i}(z, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0)) = \alpha_i q_i, Q_{a_i,q'_i}(z, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0)) = \alpha_i q'_i$ and $\forall z' \in S\setminus\{z, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0)\}, Q_{a_i,q'_i}(z, z') = Q_{a_i,q'_i}(z, z') = 0$. Let Γ be an increasing set. We distinguish three cases:

  - $z \notin \Gamma$: in this case, as $z + (-e_i + e_{p(i)}1_{[r\neq i]}, 0) \preceq_{tree} z$, we also have $z + (-e_i + e_{p(i)}1_{[r\neq i]}, 0) \notin \Gamma$, otherwise we would obtain a contradiction with the fact that Γ is increasing; hence, as both $Q_{a_i,q_i}(z, .)$ and $Q_{a_i,q'_i}(z, .)$ are zero outside of this set, we get that $\sum_{w\in\Gamma} Q_{a_i,q_i}(z, w) = \sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w) = 0$.
  - $z \in \Gamma, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0) \notin \Gamma$: in this case, as both $Q_{a_i,q_i}(z, .)$ and $Q_{a_i,q'_i}(z, .)$ are zero outside of $\{z, z + (-e_i + e_{p(i)}1_{[r\neq i]}, 0)\}$, we have that $\sum_{w\in\Gamma} Q_{a_i,q_i}(z, w) = Q_{a_i,q_i}(z, z) = -\alpha_i q_i \geq -\alpha q'_i = Q_{a_i,q'_i}(z, z) = \sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w)$.
  - $\{z, z + (-e_i + e_{p(i)}1_{[r\neq i]}, 0)\} \subseteq \Gamma$: in this case, as both $Q_{\alpha,q_i}(z, .)$ and $Q_{a_i,q'_i}(z, .)$ are zero outside of $\{z, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0)\}$, we get that $\sum_{w\in\Gamma} Q_{\alpha,q_i}(z, w) = Q_{a_i,q_i}(z, z)+Q_{a_i,q_i}(z, z+(-e_i+e_{p(i)}1_{[r\neq i]}, 0)) = -\alpha_i q_i + \alpha_i q_i = 0 = -\alpha_i q'_i + \alpha q'_i = Q_{a_i,q'_i}(z, z) + Q_{a_i,q'_i}(z, z + (-e_i + e_{p(i)}1_{[r\neq i]}, 0)) = \sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w)$.

  Hence, for any increasing set Γ, $\sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w) \leq \sum_{w\in\Gamma} Q_{a_i,q_i}(z, w)$;

- Case 3: $x_i = 0, y_i = B_i$: in this case, we have $\forall z' \in S, Q_{a_i,q_i}(z, z') = Q_{a_i,q'_i}(z, z') = 0$, and hence $\forall\Gamma$ increasing set,

$$\sum_{w\in\Gamma} Q_{a_i,q_i}(z, w) = \sum_{w\in\Gamma} Q_{a_i,q'_i}(z, w).$$

Hence, $\forall z \in S$, for any increasing set Γ, we have:

$$\sum_{w \in \Gamma} Q_{a_i,q_i'}(z,w) \leq \sum_{w \in \Gamma} Q_{a_i,q_i}(z,w)$$

and hence $Q_{a_i,q_i'} \preceq_{st} Q_{a_i,q_i}$.

$\square$

Hence, we can apply Corollary 3 to obtain the following result:

Theorem 6. *Consider an EPN network whose routing graph is a directed tree, with energy queues that are either M/M/1/B or M/M/B/B queues. If Z is a CTMC of this model with JOB probability vector q and Z' is a CTMC of this model with JOB probability vector q' such that $q \leq q'$ componentwise, then*

$$Z_t' \preceq_{st} Z_t.$$

As a consequence of the above theorem, we have the following result.

Corollary 4. *The model with full rejection of the energy packet (i.e. $\forall i \in [\![1,n]\!], q_i = 0$) is greater (in the strong stochastic sense) than the model with jump-over blocking ($\forall i \in [\![1,n]\!], q_i > 0$).*