Explicit Energy Quantification in Wireless Sensor Networks Using Petri Nets

Amel Berrachedi[1], Malika Ioualalen[2] and Ahmed Hammad[3]
[1]*Faculty of Exact Sciences and Informatics, Hassiba Benbouali University of Chlef, Chlef, Algeria*
[2]*Faculty of Computing Science, USTHB University, Algiers, Algeria*
[3]*FEMTO-ST Institute, Marie and Louis Pasteur University, Besançon, France*
a.berrachedi@univ-chlef.dz, mioualalen@usthb.dz, ahmed.hammad@femto-st.fr

Keywords: Formal Modeling, Petri Nets, Wireless Sensor Networks, Energy, Clustering.

Abstract: Recent advances in Wireless Sensor Networks (WSNs) have introduced several challenges related to their limited processing, storage, and especially energy capacity. These constraints necessitate preliminary verification to ensure the reliability required by such networks or similar systems. This paper focuses on quantifying energy consumption using a Petri Net model. To this end, we propose a formalism called Energy Petri Nets (EgPN), which explicitly models energy within sensor networks. In addition to the formal definition, we provide an algorithmic implementation enabling automated evaluation of network lifetime. We will present its applicability through a case study which is the clustering, one of the most widespread techniques for energy conservation.

1 INTRODUCTION

Since their emergence, Wireless Sensor Networks (WSNs) have increasingly penetrated the computing world. These networks consist of a base station responsible for collecting data transmitted by a set of miniature, autonomous, multifunctional sensor nodes distributed across a sensing area to monitor physical phenomena or events.

Experimental studies have shown that data transmission is the most energy-consuming task. Given that sensor nodes are typically powered by low-capacity and non-replaceable batteries, they must operate in an energy-efficient manner. The primary objective is to ensure reliable data transmission from source to destination while maximizing the network's lifetime. This challenge has led to various energy conservation techniques, such as multi-hop communication, data aggregation, and clustering. Among these, clustering has attracted particular attention: it organizes the network into hierarchical groups, with Cluster Heads (CHs) handling the most energy-intensive tasks, including intra-cluster coordination and communication with the base station.

Although these energy conservation strategies have been extensively studied and validated through simulation, what remains crucial is the ability to formally model and analyze energy-constrained systems such as WSNs. Simulations can provide valuable insights, but they often fail to capture the rigor and completeness required for verification, especially in real-world deployments. Given the critical importance of network longevity and the difficulty in replacing node batteries in remote environments, formal methods become essential. Petri Nets (PNs) are a promising tool for modeling such systems, but traditional PNs lack explicit energy representation.

This paper makes the following contribution: we introduce *Energy Petri Nets* (EgPN), an extension of PNs that integrates energy as a quantifiable and explicit parameter in the modeling of WSNs or similar systems. Beyond the theoretical formalism, we also implemented the EgPN execution scheme to carry out automated calculations of energy consumption. To illustrate the approach, static clustering is used as a case study.

The remainder of this paper is organized as follows. Section 2 discusses the application of Petri Nets in WSNs or even in energy-constrained systems. Section 3 presents the proposed EgPN model, including its formal definition and behavioral semantics. Section 4 describes the algorithmic execution scheme of EgPN, providing the basis for automated evaluation. In Section 5, we apply the EgPN model to analyze energy consumption in the static clustering scenario. Section 6 concludes the paper and outlines potential directions for extending this work.

2 RELATED WORK

Petri nets (PNs) (Peterson, 1977) are a graphical tool accompanied by solid mathematical basis, well-suited for modeling distributed systems. It is a bipartite graph composed of two types of vertices: places, which represent the system's state variables, and transitions, which represent elementary actions. The edges are directed arcs which connect either a place to a transition or a transition to a place.

Researchers have long used PNs for modeling energy consumption in WSNs, each addressing specific aspects while revealing limitations that motivate further advancements. For example in (Yang et al., 2008), the authors used Colored Petri Nets (CPNs) to quantify energy at the node level by introducing a dedicated *Energy* place for each sensor. While this solution offered a more explicit representation of energy, it significantly increased the model's complexity due to the significant increase of places and connecting arcs. Furthermore, energy values were limited to discrete integers, which poorly reflect the continuous nature of real energy usage.

The authors in (Oukas et al., 2023) proposed later an energy-harvesting-aware model based on Generalized Stochastic Petri Nets (GSPNs). Similar to(Yang et al., 2008), they introduced a dedicated *Battery* place to abstract the energy level of each sensor node. However, instead of using explicit token counts, they relied on symbolic levels and quantized energy states to simplify the model while capturing essential behaviors such as sleeping, retransmission, and energy harvesting. The model was simulated using TimeNet tool to evaluate network lifetime and performance under varying environmental conditions.

In an attempt to refine energy modeling, Li et al. (Li et al., 2014) used Queuing Petri Nets (QPNs) to represent component-level energy states such as microcontroller, transceiver, and sensors. Each energy state is modeled as a place associated with a residence time, allowing energy consumption to be calculated by multiplying the power of that state with time spent in it. System events such as reception, transmission and interruption are modeled as tokens that circulate within the network, while transitions represent the corresponding state changes. The QPME tool is used to model and simulate the energy consumption metric.

Although this approach provides detailed energy tracking and simulation, it does not embed energy directly into the system's logical behavior; that is, transitions are not conditioned on energy availability, limiting the model's capacity to express energy-constrained execution. Unlike the previous approaches, some proposed works have enhanced PN formalisms that explicitly incorporate energy consumption into the model. These contributions overcome the limitations of traditional models by enabling more accurate energy-aware analysis and performance evaluation.

Among these works, the authors in (Qin et al., 2021) introduce a dedicated modeling formalism called Hybrid Cyber Petri Net System (HCPNS) to simulate and optimize the charging process in Wireless Rechargeable Sensor Networks (WRSNs). HCPNSs extend classical Petri nets by integrating discrete event decisions, continuous energy flows and temporal dynamics enabling a detailed representation of energy behaviors in WRSNs. The sensor node's energy is explicitly represented: each place corresponds to a sensor node and holds its current energy level. Energy consumption is captured by transitions that remove energy tokens according to predefined consumption rates. Although this approach offered a realistic view of charging systems, it is specifically tailored to a master-slave charging strategy and lacks generalization to broader WSN applications or diverse routing protocols.

Another relevant contribution is presented in (Lecomte and Bonhomme, 2025), where the authors propose a solution for detecting internal transition sequences in a partially observable P-Time Labeled Petri Net (P-TLPN), optimizing its energy consumption. Each transition and each place is assigned an energy cost, reflecting actual consumption. The authors then attempt to discover possible internal sequences that reconcile the observation while minimizing total energy consumption. Finally, a recursive algorithm based on partial exploration of the state graph is used to traverse the space of compatible sequences and return those with minimal cost.

In addition to the aforementioned studies, recent research has further extended the role of PNs in energy and reliability domains. For instance, Grobelna and Szcześniak (Grobelna and Szcześniak, 2022) applied Interpreted Petri Nets to model autonomous components within electric power systems. Their work demonstrated how formal specifications of control algorithms for energy storage and stabilization services can be verified in terms of liveness, safety, reversibility, and determinism, thereby enabling early detection of design flaws. This approach highlights the potential of PNs not only for networking, but also for guaranteeing robust control in cyber–physical energy infrastructures. Similar modeling principles could be applied to WSNs, where energy management and reliable operation are equally critical.

Another work is proposed by (Nasrfard et al., 2023) where a PN-based model is introduced for optimizing inspection and preventive maintenance policies. Their stochastic formulation enabled the evaluation of different inspection intervals with respect to system availability and cost, thus providing a decision-making tool for reliability engineering. Although focused on maintenance optimization, this work confirms the versatility of PNs as a formalism capable of supporting performance and lifecycle management in complex systems. Such reliability-oriented modeling can be particularly relevant for WSNs, where node failures and limited energy resources directly impact network lifetime and service quality.

From these studies, we observe that the PNs are very well-suited for modeling energy consumption in diverse domains. Nevertheless, several limitations remain when applied to WSNs. Early approaches such as (Yang et al., 2008; Oukas et al., 2023; Li et al., 2014) introduced dedicated energy places or symbolic states, but at the cost of increased complexity or limited accuracy in capturing continuous energy evolution. More advanced extensions like (Qin et al., 2021; Lecomte and Bonhomme, 2025) explicitly incorporate energy or hybrid dynamics, yet they remain tailored to specific contexts such as master–slave charging or transition optimization rather than providing a general-purpose framework. Other contributions, including Petri Net models for energy infrastructures (Grobelna and Szcześniak, 2022) or for preventive maintenance policies (Nasrfard et al., 2023), demonstrate the formalism's applicability beyond networking, but they do not integrate energy as a first-class constraint on execution.

The gaps arising from these contributions motivate our proposal of the Energy Petri Net (EgPN), an extension of place/transition nets where transition firing depends on residual energy of sensor nodes. In an EgPN, the energy is explicitly integrated into the system's behavior by introducing it as a transition guard. This guard represents the energy required to perform the action modeled by this transition. A transition can fire only if the corresponding sensor node has sufficient residual energy. Node energy levels are represented by real-valued vectors, which are updated at each firing to ensure that the energy state evolves consistently with the system's behavior. Furthermore, EgPN supports structural analysis through incidence matrices, making the model both analyzable and scalable. By jointly capturing system behavior and energy consumption, EgPN provides a practical but simple modeling tool for WSNs. In the following section, we present the proposed EgPN model, which aims to combine expressiveness, explicit energy modeling, and formal analysis capabilities.

3 THE PROPOSED MODEL: ENERGY PETRI NET

The Energy Petri Net (EgPN) model introduces energy thresholds on all transitions. A transition can fire only if the residual energy of the executing sensor node exceeds the threshold. This enables a continuous (real-valued) and explicit energy representation. In addition, this allows to reduce the number of places and arcs in the model.

3.1 Model Formalization

We have given an intuitive idea of the proposed EgPN model. However, it is a formal model. It is therefore appropiate to provide a rigorous definition.

Definition 1. We call an EgPN model any quintuplet N=(P,T,W,e,Res) where:

- P is a finite set of places such that: $P = \{p_1,...,p_n\}$.
- T is a finite set of transitions, disjoint from P, such that: $T = \{t_1,...,t_m\}$.
- $W : P \times T \cup T \times P \rightarrow \mathbb{N}$ is the arc valuation function.
- $e : T \rightarrow \mathbb{R}$ is the function which associa tes a threshold energy with each transition.
- $Res(Res_1, Res_2, ..., Res_l)$ is the vector of residual energies in the network such that Res_i is the remaining energy of the i^{th} sensor node and l is the total number of sensor nodes in the network.

The valuation function W gathers preconditions and postconditions. They can be defined as follows:

Pre: $T \times P \rightarrow \mathbb{N}$ is the restriction of W to $P \times T$. We denote: $Pre(t, p) = W(p,t)$.

Post: $T \times P \rightarrow \mathbb{N}$ is the restriction of W to $T \times P$. We denote: $Post(t, p) = W(t, p)$.

From a graphical point of view, the conventions for places, transitions, and arcs in the EgPN model are the same as for place/transition PNs. Regarding transitions, they are weighted by the energy thresholds that are placed in parentheses next to the transitions.

In order to have a complete static description of the model, we need to complete the previous definition in order to describe the possible evolutions of the system. This is done via marking.

Definition 2. A marking M of an EgPN $N = (P,T,W,e,Res)$ is any function $M : P \rightarrow \mathbb{N}$. We call a marked EgPN any pair (N,M_0) where M_0 is the initial marking.

The markings describe the possible evolutions of the system without taking into account the energy specificities of the model. We will then introduce the notion of state of an EgPN which describes the system by taking into account these specificities.

Definition 3. We call the state of an EgPN the pair $\xi = (M,Res)$ where:

- M is the marking which indicates the number of tokens in each place.
- *Res* is the residual energies vector which indicates the remaining energy of each sensor node.

The definitions presented above pertain to static modeling, which provides a snapshot of the system at a given moment. However, sensor network-based systems are inherently dynamic and evolve over time. Therefore, it is essential to complement structural descriptions with a behavioral model that captures the system's evolution.

3.2 Behavioral Study - Dynamic Aspects

In what follows, we formally express, on the one hand, the rules for crossing a transition and on the other hand, the effects on the system after crossing this transition. This is reflected by the exploration of the marking graph.

3.2.1 Transition Firing

In an EgPN, transitions can be valid or enabled.

Definition 4. A transition t is **valid** from a marking M if and only if we have: $\forall p \in P, Pre(t,p) \leq M(p)$. We denote it $M(t \geq$.

Definition 5. In an EgPN, a transition t is **enabled** from a marking M if and only if:

- $M(t \geq$
- $e(t) \leq Res_i$ such that Res_i is the residual energy of the i^{th} sensor node performing the action t.

Definition 6. The firing of t leads the system to the state described by the marking M' defined by: $\forall p \in P, M'(p) = M(p) - Pre(t,p) + Post(t,p)$. We denote it by $M(t > M'$. Furthermore, on each node i on which the action is performed, $Res_{i_{New}} = Res_{i_{Old}} - e(t)$.

3.2.2 Marking Graph

The marking graph of an EgPN is a directed graph whose vertices are labeled by markings and whose edges are labeled by transitions.

Definition 7. We call the marking graph of an EgPN (N,M_0) the labeled graph such that:

- The initial vertex is labeled by M_0.
- If a vertex ν with label M and if t is a transition such that $M(t > M'$, then there exists a vertex ν' with label M' and an edge $M \xrightarrow{t} M'$.

To describe the energy consumption on the graph above, we introduce adaptations of the notion of marking graph. The vertices are labeled by states instead of markings.

In PNs in general, system analysis techniques rely on the initial marking. Any change in the initial marking requires the construction of the marking graph to be restarted. However, it would be desirable to analyze the system independently of the number of tokens. One solution is to study the model structural properties using analysis tools from linear algebra (Desel, 1998).

3.3 Structural Study - Linear Algebra

The structural study of the EgPN model determines the properties that are independent of M_0 and that depend only on the function W.

3.3.1 Matrices Representation

The matrix representation of EgPNs is directly derived from the graph-based structure of place/transition PNs. Accordingly, the incidence matrix, as well as the pre-incidence and post-incidence matrices, are employed to represent EgPNs.

Let $N = (P,T,W,e,Res)$ be an EgPN. It is represented using three matrices:

- The pre-incidence matrix that models the preconditions of the transitions. It is defined by: $Pre = (pre_{i,j})_{i \in 1..n, j \in 1..m}$, where $pre_{i,j} = W(p_i,t_j)$
- The post-incidence matrix that models the postconditions of the transitions. It is defined by: $Post = (post_{i,j})_{i \in 1..n, j \in 1..m}$, where $post_{i,j} = W(t_j, p_i)$
- The incidence matrix that models the effects of the transition firings. It is defined by $C = Post - Pre$.

In addition to these matrices, we need to represent the vector *Res*. We consider it as a row vector with l columns. Finally, the initial marking is described using a column vector with n rows.

3.3.2 Transition Firing Representation

To determine whether a transition is enabled, we use the matrix *Pre* and the vector *Res*. To calculate the resulting marking, we use the incidence matrix C and we must update the vector Res. However, only the incidence matrix C is needed to explore the marking graph.

Consider the transition t_j and the marking M. The transition t_j is enabled from M if and only if:

1. $\forall p, Pre(t_j, p) \leq M$, that is, if the j^{th} column of Pre is less or equal to M:

$$\begin{pmatrix} W(p_1,t_j) \\ W(p_2,t_j) \\ \dots \\ W(p_n,t_j) \end{pmatrix} \leq M \text{ Soit:}$$

$$Pre \times \begin{pmatrix} 0 \\ \dots \\ 1 \\ \dots \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ \dots \\ j \\ \dots \\ n \end{pmatrix} \leq M$$

2. $e(t_j) \leq Res_k$ where Res_k is the residual energy of sensor k wishing to perform action t_j.

We then arrive at the marking M' defined by $M' = M - Pre(t_j, p) + Post(t_j, p), \forall p$,

$$\text{that is, } M' = M + C \times \begin{pmatrix} 0 \\ \dots \\ 1 \\ \dots \\ 0 \end{pmatrix} j$$

The formal definitions and algebraic representations presented above provide the theoretical basis of the EgPN model. However, performing the evolution of the system manually from these equations quickly becomes impractical, especially for large-scale networks. To overcome this limitation, we have developed an algorithmic execution scheme that directly implements the firing rules of EgPN. This pseudocode was implemented in C programming language in order to automate the execution process and generate intermediate data stored in text files.

4 ALGORITHMIC EXECUTION OF THE EgPN MODEL

In order to complement the structural and algebraic analysis of the Energy Petri Net (EgPN), we provide an algorithmic description of its execution semantics (Algorithm 1). The pseudo-code specifies how the network evolves over time, starting from the initial marking M_0 and the initial residual energy vector Res. At each step, the set of enabled transitions is determined by verifying both the token availability in places (logical condition) and the sufficiency of residual energy (quantitative condition). If no transition is enabled, the system becomes blocked and the execution stops.

```
Data: Number of places |P| = n, transitions
      |T| = m, sensor nodes |N| = l
Data: Initial marking M0 ∈ ℕ^n, matrices
      Pre, Post ∈ ℕ^(n×m)
Data: Residual energies Res ∈ ℝ^l_≥0,
      thresholds e : T → ℝ_≥0
Data: Decision transition probabilities
      p : T → [0,1]
Data: NodePerforming: T → {1,...,l}
Result: Number of rounds, Residual energies
        Res
M ← M0;    C ← Post − Pre;
 round ← 0;
while true do
    Enabled ← ∅;
    foreach transition t_j ∈ T do
        if Pre(:,t_j) ≤ M then
            (in Pre, the column of t_j only)
             k ← NodePerforming(t_j);
            if Res_k ≥ e(t_j) then
                add t_j to Enabled;
            end
        end
    end
    if Enabled = ∅ then
        break;
    end
    DecisionSet ← ConflictingFrom-
      SamePlace(Enabled, Pre);
    if DecisionSet ≠ ∅ then
        t_j ← ChooseByProbabil-
          ity(DecisionSet, p);
    end
    else
        t_j ← ChooseDeterminis-
          tic(Enabled);
    end
    k ← NodePerforming(t_j);
    M ← M + C(:,t_j);
    (in C, the column of t_j only)
     Res_k ← Res_k − e(t_j);
    if M = M0 then
        round ← round + 1;
    end
end
return round, Res;
```

Algorithm 1: General EgPN Execution: Network Evolution.

Decision Transitions and Probabilities. In our setting, probabilities are used only at decision points, i.e., when a place feeds two or more outgoing transitions (a structural conflict). In such a case, exactly one transition must fire, and the choice among the conflicting transitions follows the prescribed probabilities $p(t)$. Outside of these local conflicts (e.g., when several enabled transitions do not compete for the same tokens), no stochastic choice is applied.

Priorities. We deliberately do not model priorities between concurrently enabled transitions that are not in structural conflict. When several such transitions are enabled simultaneously, a simple deterministic tie-breaking policy (e.g., fixed order) may be used; exploring priority schemes is left out of the present work.

Network Lifetime. This algorithmic scheme allows one to simulate the execution step by step, count the number of returns to the initial marking, and monitor the residual energy per node. Knowing the number of completed rounds (returns to M_0) and assuming the duration of one round is known, the network lifetime is calculated as the number of rounds multiplied by the duration of one round.

To better highlight the evolution and usefulness of the EgPN model, we apply the proposed algorithm to static clustering in the next section.

5 CLUSTERING MODELING USING EgPN: A CASE STUDY

To demonstrate the applicability of the proposed EgPN formalism, we present a case study based on clustering, one of the most widely adopted energy-conservation techniques in WSNs. To make the model operational, it is therefore necessary to specify a set of initial assumptions.

5.1 Initial Assumptions

In this case study, we admit the parameter values provided in the Table 1 and considered by (Heinzelman et al., 2000). Since we focus only on energy parameter, we admit:

- The network lifetime ends when the first sensor node depletes its energy while performing an action. Fault tolerance (the ability to keep operating despite node failures) is not considered and is left for future work.
- Packet transmissions are assumed to be always successful. Packet loss tolerance is not addressed at this stage.

Table 1: The Network's Parameters.

Parameter	Value	Signification
$d_{MBR,CH}(m)$	18	Average distance between members and their CHs
$d_{CH,BS}(m)$	100	Average distance between CHs and the base station
k(bit)	2000	Size of a data packet
E_0(mJ)	50	The initial energy of each sensor node
$E_{elec}(nJ/bit)$	50	Energy required for transmission/reception
$E_{amp}(pJ/bit/m)$	100	Energy needed to amplify the signal during a transmission
$E_{sense}(nJ)$	3	Energy needed for sensing
$E_{aggr}(nJ/bit)$	5	Energy required for aggregation

5.2 EgPN Model for Static Clustering

To reduce the complexity of our model, we consider a cluster of three nodes. This approach can be generalized to a cluster of 20 nodes. However, due to space limitations, this generalization is not addressed in the present work and will be presented in future research. The EgPN model corresponding to a cluster of size 3 is shown in figure 1. The definitions of the places and transitions are provided in Tables 2 and 3, respectively.

Table 2: Definitions of the Places of the Static Clustering Model.

Place	Description
MBR_i	Initial state of the i^{th} member
Ready-S_i	The i^{th} member is ready to send the data packet to its CH
Transit-D_i	The i^{th} member sent its data package to its CH
D-MBR_i	The CH received the data packet from its i^{th} member
F-D	The aggregated data packet

After injecting the previously presented parameters into the C mini-tool, the energy consumption was recorded round by round, we summarize here the final results presented in Table 4.

We obtained a network lifetime of 21 rounds for 3 sensor nodes which is quite short. This lifetime is expected to be even shorter when the number of nodes increases, for example to 20 nodes. This limitation is

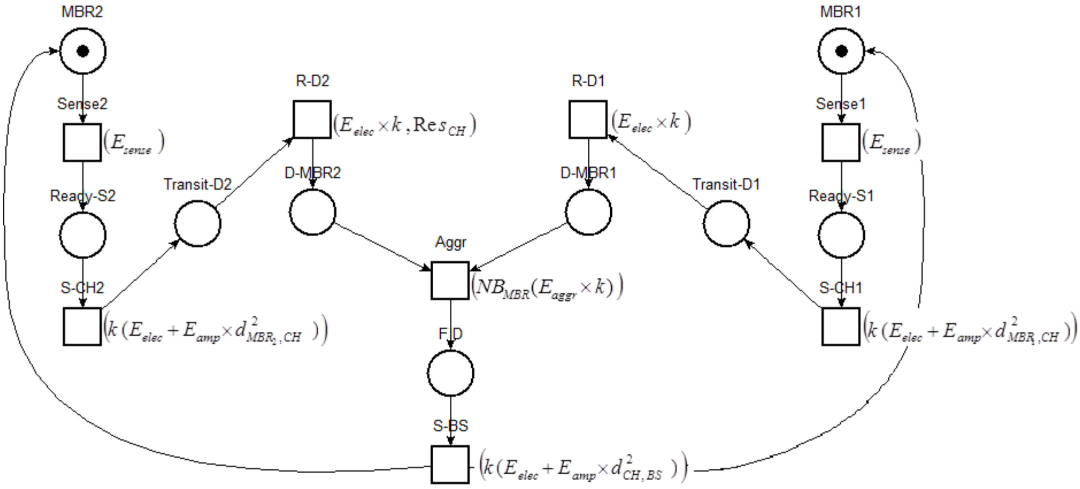

Figure 1: EgPN model for the static clustering.

Table 3: Definitions of Transitions of the Static Clustering Model.

Trans.	Description	e(transition)
$Sense_i$	$Member_i$ senses information	$E_{sense} = 3\times10^{-6}mJ$
S-CH_i	$Member_i$ sends its packet to its CH	$k(E_{elec}+(E_{amp}\times d^2_{MBR_i,CH})) = 0.1648mJ$
R-D_i	CH receives the packet from its $member_i$	$(E_{elec}\times k)=0.1mJ$
Aggr	CH performs the data aggregation	$NB_{MBR}(E_{aggr}\times k) = 0.02mJ$
S-BS	CH sends the final packet to BS	$k(E_{elec}+(E_{amp}\times d^2_{CH,BS}))=2.1mJ$

Table 4: Energy Consumption Results for the Static Clustering.

	CH	MBR_1	MBR_2
Consumed Energy (mJ)	48.94	3.63	3.63
Remaining Energy (mJ)	1.06	46.37	46.37

mainly due to the use of static clustering, where the same nodes remain CHs and consume energy faster. For this reason, many researchers have proposed dynamic clustering approaches that allow the rotation of CH roles in order to better balance energy consumption and extend the network lifetime.

6 CONCLUSION

In this work, we proposed a model called Energy Petri Net (EgPN), an extension of classical PNs that explicitly integrates energy as a quantifiable parameter. This formalism offers a rigorous way to model and analyze energy-constrained systems such as WSNs.

Compared to classical simulation approaches, EgPN has several advantages. It relies on a formal mathematical foundation, which allows structural analysis through incidence matrices. In addition, both behavioral and energy properties can be studied within the same framework.

We also introduced an algorithmic execution scheme, which makes it possible to automatically evaluate energy consumption and estimate network lifetime. The case studies on static clustering showed that EgPN can capture both deterministic and probabilistic behaviors, and that the obtained results are consistent with the initial assumptions.

However, some limitations remain. Concurrent transitions are resolved only by a simple sequential rule without taking into account their priorities. Other important performance metrics, such as packet loss, fault tolerance, or the case where a transition can be executed by several nodes, are not yet taken into account.

We plan to address these issues in future work. First, it would be useful to enrich the execution scheme with a FIFO structure in order to generate the marking graph and perform qualitative analysis before moving to performance evaluation. Second, the mini-tool implemented in C programming language could be improved by adding a graphical interface to make scenario configuration and results collection easier. As a final perspective, we think to integrate additional metrics such as: packet loss, priorities between concurrent transitions, or resilience when some nodes deplete energy too early, will provide a more complete assessment of WSN protocols. In the longer term, we also plan to apply EgPN to other energy-constrained systems, which would confirm its generality and relevance as a formal modeling tool.

REFERENCES

Desel, J. (1998). Basic linear algebraic techniques for place/transition nets. In Reisig, W. and Rozenberg, G., editors, *Lectures on Petri Nets I: Basic Models: Advances in Petri Nets*, volume 1491 of *Lecture Notes in Computer Science*, pages 257–308. Springer, Berlin, Heidelberg.

Grobelna, A. and Szcześniak, R. (2022). Application of interpreted petri nets in control of autonomous components in power systems. *Applied Sciences*, 12(10):4772.

Heinzelman, W. R., Chandrakasan, A., and Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS)*, volume 2, page 8020, Maui, HI, USA.

Lecomte, C. and Bonhomme, P. (2025). Energy optimization for p-time labeled petri net systems with unobservable transitions. *International Journal of Dynamics and Control*, 13(5):202.

Li, J., Zhou, H. Y., Zuo, D. C., Hou, K. M., Xie, H. P., and Zhou, P. (2014). Energy consumption evaluation for wireless sensor network nodes based on queuing petri net. *International Journal of Distributed Sensor Networks*, 10(4):262848.

Nasrfard, F., Mohammadi, M., and Karimi, M. (2023). A petri net model for optimization of inspection and preventive maintenance rates. *Electric Power Systems Research*, 216:109003.

Oukas, N., Boulif, M., Campo, E., and Van Den Bossche, A. (2023). A new generalized stochastic petri net modeling for energy-harvesting-wireless sensor network assessment. *International Journal of Communication Systems*, 36(11):e5505.

Peterson, J. L. (1977). Petri nets. *ACM Computing Surveys*, 9(3):223–252.

Qin, H., Zhao, B., Xu, L., and Bai, X. (2021). Hybrid cyber petri net modelling, simulation and analysis of master-slave charging for wireless rechargeable sensor networks. *Sensors*, 21(2):551.

Yang, S., Youssef, H., and Zhang, L. (2008). Modeling energy consumption in wsns using colored petri nets. *Simulation Modelling Practice and Theory*, 16(2):168–182.