HINTS: a Methodology for IoT Network Technology and Configuration Decision

Samir Si-Mohammed[a,b,*], Thomas Begin[a], Isabelle Guérin Lassous[a], Pascale Vicat-Blanc[b,c]

[a] *Univ Lyon; ENSL; UCBL; CNRS; INRIA; LIP; F-69342; LYON Cedex 07; France*
[b] *Stackeo; Lyon; France*
[c] *INRIA; France*

Abstract

The last decade has seen the explosion of the Internet of Things (IoT), which is enabling a range of new applications based on the connection of physical objects to the Internet. The growing diversity of IoT connectivity technologies is bringing new challenges to IoT solution designers. Indeed, it is increasingly difficult to choose and configure the network technology for a given use-case. In this article, we formalize and investigate the design optimization problem for selecting and configuring the IoT connectivity technology of an application that can evolve over time. Finding the right abstractions and the good balance between performance and evaluation complexity to compare networking options is a key research challenge. To address this problem we propose to separate the concerns of IoT application architects from those of network experts and to provide a methodology, HINTS, to help designers in making customized decision. HINTS combines IoT application requirements and goals abstraction, IoT network modeling, discrete-event network simulation and a multiple attribute decision making method. The application of the methodology on three use-cases highlights how it helps in (i) selecting the best network technology option, (ii) defining an appropriate configuration and (iii) anticipating the behavior when device density or workload intensity scales up. The main contribution of this paper is to propose the first formal approach and associated algorithm to automatically optimize the design of the IoT connectivity of an application. Results show that it can yield up to a factor two improvement in the solution performance.

Keywords: Internet of Things, Network technology, Performance, Methodology, Multi-criteria Decision, Selection, Configuration, Simulation.

---

*Corresponding author
Email address: samir.si-mohammed@ens-lyon.fr (Samir Si-Mohammed)

1. Introduction

1.1. Motivations and needs

For about fifteen years, the Internet of Things (IoT) has evolved from a research concept to reality. Embedded in the operations of industries and organizations as well as in our homes and everyday lives, it has become vital and transparent. IoT devices are everywhere. Examples include asset trackers that show the location of our shipped packages and warn us of potential delays in their delivery time; smart meters that monitor energy consumption; sensors that detect water leaks or air pollution; remote control systems that automate manufacturing equipment activity to name a few. They power a range of new applications and services with real-time data and commands. This fusion of the physical world and the virtual world can increase the operational efficiency of organizations, but also ensure the sustainability of our world and the health of its inhabitants.

Dozens of new connected products and pre-packaged or tailored solutions are launched everyday. Analysts predict that by 2030, the IoT could enable from 5.5 to 12.6 trillion dollars in value globally, including the value captured by consumers and customers of IoT products and services [1]. For example, shipments of asset trackers will grow by more than 50 percent annually through 2024 [2].

The technologies required to implement end-to-end IoT solutions are profusely available. From an architecture view point, the network layer, as defined by the ISO/OSI model of the IoT communication system, interconnects the subsystems composing the end-to-end IoT solution. At the edge of this interconnection, the physical IoT Network is the critical sub-system, which enables the connectivity of the IoT devices to the Internet. IoT Network technologies keep evolving to address the specific connectivity and communication requirements of an increasing number of IoT devices and applications. Several technologies such as Wi-Fi, BLE, Zigbee, IEEE Std 802.15.4 TSCH, Wi-Fi HaLow, 6LoWPAN, LPWAN (LoRaWAN, Sigfox, NB-IoT, LTE-M), mesh IoT architectures like Wirepas or MYTHINGS as well as 5G have been proposed and are regularly improved to tackle the heterogeneity of requirements.

However, the proliferation of candidate technologies, the inter-dependency and specificity of their respective setting parameters and the multi-criteria nature of the decision problem, make the design and configuration tasks increasingly difficult. IoT is indeed a complex technology, which heavily depends on communication performance and quality of service. For example, in Industry 4.0 or smart grid applications, communications require high reliability and low latency, whereas smart building solutions need flexibility and scalability. The authors of [3] have identified the most important properties that IoT-systems can offer: Evolution and Interoperability, Availability and Resilience, Trust, Security and Privacy, Performance and Scalability. In the design phase ([4, 5]) it is critical to select a wireless network technology that matches the performance and cost requirements of the targeted application but also the growing ambition of the IoT product owner. In the deployment phase, the appropriate configuration of the application communication parameters as well as the network settings have also a tremendous impact in terms of QoS and energy efficiency of the whole solution. During the exploitation of a deployed solution, the connected service can evolve and the application developer may have to increase the traffic generated by or sent to the connected devices in a way that can heavily affect the performance of the end-to-end application and devices battery lifetime. As many IoT network technologies and settings could fit a given use-case, IoT solution architects would like to test and compare several of them to select and configure the most adapted network to the targeted application.

Selecting the best network technology alongside with its topology and configuration is critical for the success of an IoT solution. Recent studies (e.g., [6, 7]) have shown that almost 75% of IoT projects, be it in the US, UK, or India, were deemed failures and that 30% of IoT projects actually failed to move beyond the proof-of-concept stage. While customers see real value in deploying IoT, many industrial companies and projects are lagging behind – for example up to 70 percent of industrial companies projects end up in "pilot purgatory" [8]. The skills shortage and the difficulties in navigating the technological ecosystem are part of the barriers that explain these failures. The profusion of possibilities often results in non-decision, non-optimal choices, excessive total cost of ownership and, ultimately, project failure. For instance, the success of an IoT project can be jeopardized due to an insufficient budget or wrong technological decisions such as an inadequate network technology.

1.2. Complexity of IoT network technology evaluation

Undoubtedly, testing the sensing and network technologies with real and in-context deployments is key for assessing the functional feasibility. Verifying the compatibility of sensors with a given network technology is crucial, in addition to the validation of the quality of collected and transported data. Understanding how the connected solution impacts the current human processes is also of great importance. In general, these are the goals of a pilot project. During this phase, various hardware and physical parameters can be evaluated and calibrated to find the right adjustments. These benchmarks serve as ground truth. Unfortunately, they only give a narrow and insufficient picture of the future reality as the scope of the test is typically limited to a few devices.

Imagine that as the architect of a pre-packaged tracking solution for construction vehicles, you have selected a given network technology. You have based your choice on the experience you have on the technical specification or on a proof-of-concept made with an early adopter. Later, in the course of your business, one of your customer discovers that the location accuracy is far too low for their needs due to limited bandwidth or coverage. These technical limitations prevent the tracking of vehicles when they pass through certain geographic areas. Your product-solution is functional in some restricted contexts but you end up by addressing a limited market, which could potentially impact the success of your business. To make a wiser and safer decision, you would have needed to have a large-scale vision and a long-term perspective at design time.

However, making large and long experiments was out of your reach because the cost, time and complexity of installation were prohibitive. Additionally, IoT network technologies and architectures are constantly evolving – 5G, satellite and edge settings to name a few – adding to your confusion.

For optimal and cost-efficient decision-making computer-based simulation has been proven to be one of the most valuable aids for systems design and evaluation. However, the use of this approach remains limited in the IoT infrastructure field. The main reason for this is that emerging wireless IoT networks are increasingly complex and the user-friendly capabilities provided even by the most advanced simulation tools are not sufficient to cope with such complexity [9]. Furthermore, it appears that, for a specific application, the number of IoT network settings to be explored can be very large, making simulation campaigns complex. Most organizations do not have the talents, nor the budget to afford such time-consuming studies.

We think that a decision-support methodology and an associated tool to systematize the evaluation process would be of great help. This would enable to objectively compare technology candidates and to deliver key performance indicators (KPIs) to support decision making at the IoT solution's design, deployment or exploitation phases. The ultimate goal is to future-proof and select the most appropriate network technology for a new application development or an adequate configuration for a new deployment of a pre-packaged IoT solution. However, finding the right abstractions that lead to a good balance between performance accuracy and computational complexity when comparing networking options remains an open and challenging issue.

1.3. Contributions and Outcomes

In this article, we focus on the properties Performance and Scalability ([3]) of the IoT network which critically impact the ability of the IoT solution to predictably execute within its mandated performance profile and to handle increased processing volumes in the future if required. We formalize and investigate the design optimization problem for selecting and configuring the IoT connectivity technology of an application that can evolve over time. To address the abstraction complexity problem, we propose to separate the concerns of IoT designers from those of network experts. We leverage a fine balance between performance accuracy and computational complexity to provide a methodology, HINTS, that combines IoT application requirements, goals formulation, IoT network modeling, discrete-event network simulation, and a multiple attribute decision-making method. The main contribution of this paper is to propose the first formal approach and associated algorithms to automatically optimize the design of the IoT connectivity for a given application. Our numerical results show that HINTS can lead up to a factor of two improvements in the solution performance. HINTS can prevent designers and architects from wasting time and money on complex experimentation and taking important risks.

The novel contributions of the paper are as follows:

- **A comprehensive modeling approach** to abstract the communication requirements of any IoT application and to capture the specificity of any IoT network technology.
- **A pre-selection method** to filter network candidates meeting the communication requirements of a given application.
- **A systematic evaluation method** to estimate the performance but also to predict long-term behavior (scalability) of a set of selected network candidates.
- **A multi-criteria decision method** to customize and assist the final decision process.

The main contribution of this paper is to propose the first formal approach and associated algorithm to automatically optimize the design of the IoT connectivity of an application. Section 2 discusses the related work. In Section 3 we formulate the problem of IoT network technology design and configuration. In Section 4, we present our solution to the problem. Section 5 is devoted to the illustration of the implementation of the methodology on three use-cases. Finally, Section 6 concludes this paper.

2. Related Work

The problem of network selection, taken in its broadest sense, has attracted much attention since the rapid deployment of wireless networks.

When the choice for the most adequate network technology must be made for a forthcoming IoT solution, IoT surveys are a valuable source of information regarding the modulations in use, the channel bandwidth, the maximum payload sizes, and the authentication and encryption support. Surveys cover IoT communication in general (*e.g.*, [10, 9]), or are more specifically devoted to Low-Power Wide Area Networks (LPWAN) technologies (*e.g.*, [11, 12, 13, 14]). Either way, these surveys have limitations: they provide coarse-grained information for Key Performance Indicators (KPIs), representing networking performance metrics, often limited to best and worst-case values, independent of the targeted IoT applications. Yet, in IoT, the choice of the right network technology is strongly tied to the specific requirements of the application, the network topology, the environment as well as the resources embedded within end-devices. For instance, the theoretical capacity of network technologies can be misleading as in practice, this value will often not be reached because of (i) the channel errors and/or interference, and (ii) the contention resulting from end-devices attempting to access the radio channel at the same time.

A number of works have conducted performance studies in a bid to compare the efficiency of two or more network technologies at supporting an IoT application. Typically, they consider a specific scenario and evaluate the associated performance using simulations or real experiments (*e.g.*, [15, 16]). In [16] the authors assess the relative merits of NB-IoT, SigFox, and LoRaWAN in covering the needs of smart water grids using the simulator ns-3. While they conclude own the superiority of NB-IoT, their study does not take into account major KPIs such as latency, cost, range, and energy consumption. [17] compares between Wi-Fi HaLow, LoRaWAN and NB-IoT for smart city applications, using simulation. In [15], the authors use simulations to compare the coverage and capacity of SigFox, LoRaWAN, GPRS, and NB-IoT at meeting the needs of a large-scale IoT deployment. While all technologies were found able to cover most of the needs in terms of coverage for outdoor communications, their results show that NB-IoT, and to a lesser extent SigFox, outperform the others for indoor communication. However, their results do not consider energy- and delay-related KPIs. All these works do not provide any generic tool for the selection and the configuration of network technologies. To fulfill this gap, we think that additional steps are required, such as the modeling of the targeted IoT application and the network alternatives, the application of an evaluation framework to evaluate the KPI values obtained with the network alternatives, and the use of a comparison method to rank the different network alternatives and to identify the best one. In [18], a framework for the IoT network technology evaluation is proposed. It gives special attention to the energy consumption efficiency, but does not provide any comparison and decision support. In [19], the authors propose a 2-step methodology to guide IoT users to choose the appropriate network technology for their needs. First, they use a questionnaire to eliminate network technologies based on the mismatch between the application requirements and the network technology characteristics. Then, they propose an evaluation of the main cost components to find the most economical network technology. Overall, their solution can be viewed as a solid step towards an automatic selection method for the choice of an IoT network technology. However, the considered values for KPIs are constant (when they should vary with the scenario under consideration) and the relative merits of IoT network technologies are compared only through their financial cost.

IoT testbeds are leveraging real IoT devices and thus represent great options to evaluate IoT applications under real-world conditions. They often come with API facilitating the design of the experiments and the processing of the collected data. Examples of open research testbeds include FIT loT-LAB [20], SmartSantander [21], COPELABS IoT [22], FIESTA-IoT [23]. However, open IoT testbeds have their own limitations. Their instrumentation may be incomplete when, for instance, measurements of the delay and energy consumption are required. The use of an IoT testbed also implies constraints on the considered topology and scale. This may pertain to the distance between end-devices and their gateway, the maximum number of available IoT devices, and the environment which is typically indoors and not under the control of the researchers performing experiments.

In terms of decision support, studies have mostly focused on the dynamic interface selection (aka vertical handoff - VHO) with the goal of favoring the performance of end-users [24, 25, 26]. These studies naturally lead to dynamic multi-criteria decision problems where a utility function must be cautiously devised using algorithms such as Simple Additive Weighting (SAW), Weighted

Product Methods (WPM), and Technique for Order Preference Similarity to Ideal Solution (TOPSIS) [27]. To handle the exploration-exploitation dilemma inherent to the dynamic selection of interfaces, researchers have resorted to various mathematical approaches such as combinatorial optimization [28, 29], fuzzy logic [30, 31], Markov Decision Processes [32, 33], game theory [34, 35], and machine learning technique such as the multi-armed bandit framework [36, 37]. All these works suppose that a heterogeneous wireless network has been put in place and configured so that probing tests can be performed before selecting the most adequate network interface for each end-device. This is in contrast with our study for the static selection of the network technology, which we assume has not been yet deployed, nullifying the possibility of collecting performance probing.

Overall, a general IoT network technology decision method, for the design, configuration, and exploitation phases of an end-to-end IoT solution that takes into account both the key aspects of the application and of the network with regard to KPIs, is still missing. The goal of this paper is to fill this gap. Table 1 provides a summary of the related works, and how HINTS differs from the existing works. Note that we use "N/A" to denote papers in which the authors do not disclose the method used to evaluate the different network technologies. The column named "Application-driven" specifies whether the authors consider or not the specificity of the IoT application. Finally, the column named "Decision support" states if the authors introduce an algorithm to compare and select the network technology and selection (for instance, using MADM methods) of the network technologies. Table 1 shows that HINTS stands out from the other methods by being an application-aware method and by providing an automatic selection mechanism at the same time. As for the KPIs, HINTS deals with the same set as many other existing works. This set includes the most important KPIs in the IoT networking field. Regarding the evaluation method to obtain the KPIs, most existing works ([10, 11, 12, 13, 14, 19, 24, 25, 26]) do not detail their way of evaluating the different network technologies. A number of works ([15, 16, 17, 18]) refer to discrete-event simulation, as HINTS does.

Reference	Considered KPIs				Method to evaluate KPIs	Application driven	Automatic Selection
	Message delivery	Message latency	Energy consumption	Cost			
Kanuch et al. (2020) [10]	-	-	✓	-	N/A	-	-
Sinha et al. (2017) [11]	-	✓	✓	✓	N/A	-	-
Ikpehai et al. (2018) [12]	✓	✓	✓	-	N/A	✓	-
Mekki et al. (2018) [13]	✓	✓	✓	✓	N/A	-	-
Mekki et al. (2019) [14]	✓	✓	✓	✓	N/A	-	-
Vejlgaard et al. (2017) [15]	✓	-	-	-	Simulation	✓	-
Lalle et al. (2019) [16]	✓	-	-	-	Simulation	✓	-
Verhoeven et al. (2022) [17]	✓	✓	-	-	Simulation	✓	-
Si-Mohammed et al. (2022) [18]	✓	✓	✓	-	Simulation	✓	-
Vannieuwenborg et al. (2018) [19]	-	-	-	✓	N/A	-	✓
Bari and Leung (2007) [24]	✓	✓	-	✓	N/A	-	✓
Bari and Leung (2007) [25]	✓	✓	-	✓	N/A	-	✓
Senouci et al. (2016) [26]	✓	✓	-	✓	N/A	-	✓
HINTS	✓	✓	✓	✓	Simulation	✓	✓

Table 1: Comparison of HINTS with the related work.

3. Problem formulation

3.1. General description

In general, systems stakeholders have non-functional requirements such as performance, security, or scalability [38] which have to be taken into account and implemented as properties when designing systems. Here, we focus on the Performance and Scalability of the physical IoT network, which denotes the ability of the system to predictably execute within its mandated performance profile and to handle increased processing volumes in the future if required.

We formulate this design optimization problem for IoT network performance and scalability that we are addressing as follows. Let us consider:

- An IoT application A, with its set R of characteristics and communication requirements.
- A set T of network technologies candidates.
- For each network technology T_i in T, a set C_i of possible network configurations.
- A set K of key performance metrics or KPIs that characterize the behavior of an application A on a network technology T_i with C_i.
- A set G of performance goals, defined as thresholds targeted by the application designer for each KPI.

The decision problem consists in finding the network technology T_d in T and the associated network configuration C_d^k (k-th configuration of the network technology T_d) that fit the application requirements R and best match the performance goals G of the application A, in terms of KPIs K.

The first step is to abstract and formalize the communication requirements of the targeted application and the characteristics of any network technology. Then, the good match of a network technology and its configuration to this context has to be evaluated. For this, a systematic analysis of the behavior of the application A on network technologies candidates is needed. Inappropriate candidates should be dismissed and the remaining ones compared to provide the insights required to make the final choice.

To explore the various parameters that mostly affect the application performance, the key characteristics of an IoT application as well as those of an IoT network technology must be defined. This abstraction drastically reduces the complexity of an IoT scenario evaluation. The potential evolutions of the application and of its load on the network have to be taken into account, as well as the scalability of the deployment in terms of number of connected devices. Certain other aspects considered to be of secondary importance for the purposes of this decision process can be overlooked.

In the following, we detail the application abstraction, the KPIs definition and the network technology modeling. Table B.16 lists all the notations used in the HINTS solution.

3.2. Application abstraction

First, let us model an IoT application. The principle of the application abstraction consists in characterizing the load imposed on the network by the application scenario over the time. This load is a function of the number of end-devices and the individual traffic they are going to send. We only consider static scenarios here. To study the evolutivity and scalability of the solution, the minimal and maximal values expected for the different selected parameters have to be specified. Then the factors that characterize the environmental conditions which also impact wireless communications performance are added. The specific communication requirements of an IoT application are defined by the end-devices that are communicating, by the workload they impose on the network and their physical environment as defined below. Concerning the end-devices, we focus on:

(i) The minimal number of end-devices that will be connected.

(ii) The maximal numbers of end-devices that could be ultimately connected.

(iii) The battery capacity of the end-devices.

For the workload we have:

(iv) The traffic direction (downstream and/or upstream traffic).

(v) The message size.

(vi) The minimal message frequency.

(vii) The maximal message frequency, that could be ultimately submitted.

The type of physical environment and deployment characteristics as follows:

(viii) The deployment scope, which is represented by the maximum distance expected between two end-devices.

(ix) The environment, which defines the radio conditions in which the IoT application is deployed.

(x) The expected lifetime of the deployed IoT solution must also be defined.

For the environment, we consider two cases: Indoor or outdoor, where the latter can be either (a) rural, (b) suburban or (c) urban. Inspired by [39], we propose to associate a loss propagation model to each environment type to characterize this environment as shown in Table 2.

We consider that the knowledge of the ten parameters (i) to (x) is sufficient to characterize the targeted application scenario in the case of static end-devices for the network decision process. In the case of mobile end-devices, the mobility model will have to be specified. We do not consider this type of scenario in this paper. Figure 1 illustrates the set of parameters considered by HINTS to model and capture the main characteristics of an IoT application deployed in a rural environment.

Environment Type	Propagation Model [39]
Indoor	HybridBuildings
Outdoor Rural	OkumuraHata
Outdoor Urban	COSTHata
Outdoor Suburban	LogDistance

Table 2: Environments and their propagation model.

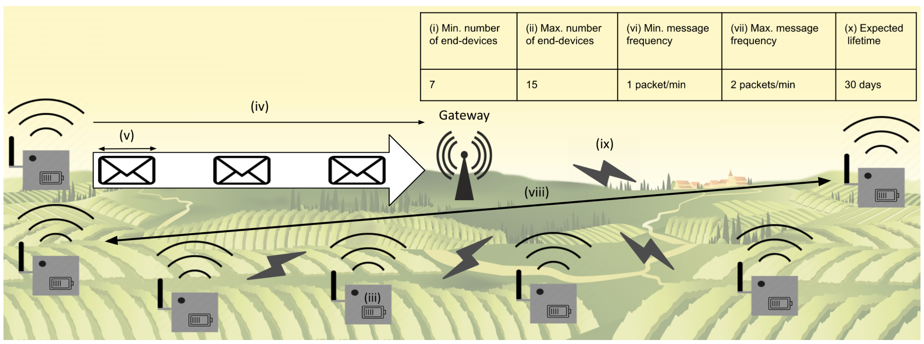

Figure 1: Abstraction of IoT Application whose characterizing parameters are: (i) Min. number of end-devices, (ii) Max. number of end-devices, (iii) End-device battery capacity, (iv) Traffic direction, (v) Message size, (vi) Min. message frequency, (vii) Max. message frequency, (viii) Deployment scope, (ix) Radio environment and (x) Expected lifetime.

3.3. KPIs definition

A lot of metrics can reflect the performance of a communication solution within an IoT context. The five major KPIs we suggest are:

- **Message delivery rate**, which represents the ratio of successfully received messages over all sent messages.
- **Energy consumption**, which is the total amount of energy consumed by end-devices to exchange messages for the lifetime of the project. In this work, we do not consider the energy consumption due to the sensing/actuating, since it is often much lower than the transmission costs in IoT systems [40]. In case they cannot be neglected, HINTS can easily be adapted to account for these sensing/actuating costs.
- **Battery lifetime**, which is the amount of time that the end-devices batteries can last before their energy get depleted (note that we consider the first energy depletion time at an end-device as the battery lifetime of the whole system).
- **Message latency**, which is the average time that a message takes to travel from the source to the destination.

- **Cost**, which represents the financial cost for deploying and maintaining the considered network technology given a selected number of gateways and network configuration for the lifetime of the project.

Note that other KPIs such as security, deployment complexity or environmental impact can also be included if the case study of interest requires it.

We categorize these KPIs in two classes: (i) Without threshold and (ii) with threshold. The former class comprises KPIs for which the goal is defined as "the more (or the less), the better". This can be the case of the cost KPI for example. The second class includes KPIs for which a goal can be quantified as a threshold. This means that there is no need to go beyond (or below) a given threshold, as further discussed in Section 4.6.

3.4. Network technology modeling

Finally, we need a model for describing, exploring and comparing a large set of network technologies. For each network technology, we identify generic and specific parameters. Generic parameters, such as the maximum data rate (or bandwidth), the frequency band, the topology type characterize any network technology. Specific parameters are dependent on each network technology. For instance, in the case of LoRaWAN, the specific parameters include the spreading factor (SF), the coding rate and the type of traffic (unconfirmed or confirmed). We note that some parameters are easily configurable by architects or by software (*e.g.*, SF for LoRaWAN) while others tend to be less tunable or simply out of reach for the architects (*e.g.*, transmission power for LoRaWAN or MCS in Wi-Fi). Table 3 represents an example with the generic and the specific parameters of the LoRaWAN network technology.

Network technology	LoRaWAN
List of generic parameters	• Data rate • Frequency band • Topology type
List of specific parameters	• SF • Coding Rate • CRC • Type of Traffic

Table 3: Network technology modeling for LoRaWAN.

The application abstraction, the KPIs and the network technology models enable us to formulate the decision problems in detail as follows:

Algorithm 1 **Problem formulation for the IoT network technology selection problem**

1: **Inputs**:

- **Application**:
  $R = [R_1, \ldots, R_k]$; Application requirements (*e.g.*, min and max number of devices, size and frequency of messages, etc.).
- **KPIs**:
  $K = [K_1, \ldots, K_n]$, KPIs (*e.g.*, Message delivery rate, Battery lifetime, Message latency, Cost, etc.).
- **KPIs performance goals**:
  $G = [G_1, \ldots, G_n]$, $G_i \in \mathbb{R}_+$; KPIs performance goals (or KPI thresholds) ($G_i < 0$ means that the i-th KPI is without threshold).
- **KPIs weights**:
  $W = [W_1, \ldots, W_n]$, $W_i \in \mathbb{R}_+$, $\sum_{i=1}^{n} W_i = 1$; Weights attributed to each selected KPI.
- **Set of network technologies candidates**:
  $T = [T_1, \ldots, T_m]$; IoT network technologies (*e.g.*, LoRaWAN, LTE-M, Wi-Fi, etc.).
- **Network configuration parameters**:
  $C = [C_1, \ldots, C_m]$; Network configurations per network technology (*e.g.*, Spreading Factor for LoRaWAN, Modulation and Coding Scheme for Wi-Fi, etc.).

2: **Outputs**:

- **Decision D1**: Select the network technology T_d that best matches the KPIs goals.
- **Decision D2**: Select the network configuration C_d^k for the chosen network technology T_d which best matches the KPIs goals.
- **Decision D3**: Select the minimal number of gateways g_d for the chosen network technology T_d which best matches the KPIs goals.

Note that in this work, the focus is on network technologies based on a star topology. Mesh networks as well as hierarchical network interconnections leveraging routing protocols such as RPL [41], represent alternative architectures for IoT connectivity. They raise interesting additional decision questions such as determining the most power-efficient routing and load balancing strategies that will be examined in future works.

4. Proposed Methodology

4.1. Overview of the methodology

The HINTS methodology targets simplicity, efficiency and risk limitation to address the IoT network decision problem formulated in Algorithm 1. To attain these objectives, the HINTS methodology is divided into two parts: (i) the network modeling part, which addresses the concern related to network experts and (ii) the application-driven decision part, which addresses the needs of application architects.

The network technology modeling part consists in abstracting and quantifying the relevant parameters of network technologies. This can be done by network experts, on the basis of the technical specification documentation and experimental evaluation. Table 4 gives the network technology models, considered by HINTS, for LoRaWAN, 5G mmWave, Wi-Fi HaLow, Wi-Fi and 6LoWPAN. These models can then be shared by the community and exploited by application architects to make data-driven decisions.

The application-driven decision part of HINTS, illustrated by Figure 2, is divided into 5 steps as follows:

1. **Application modeling**, where the value of the application requirements, listed in Section 3.2, KPIs performance goals and weights are defined.
2. **Pre-selection**, where network technologies candidates are filtered based on their technical specifications and on the application requirements.
3. **Scenario design**, where what-if scenarios, integrating the application with remaining network technologies, are designed.
4. **Evaluation**, where the what-if scenarios are instantiated and executed on an evaluation environment and the KPIs of each what-if scenario are obtained.
5. **Decision**, where what-if scenarios are ranked and the best network technology and its associated configuration are identified via a multi-criteria decision-making approach.

4.2. Application modeling

The **application modeling** step aims at:

- Quantifying the application requirements (an example for a smart building use-case is given in Table 5).
- Specifying the targeted KPIs and defining their performance goals.
- Attributing weights to KPIs.

An example of the KPIs, thresholds and weights for a smart building use-case is provided in Table 6. Recall that $G_i < 0$ means that the i-th KPI is without threshold.

Network technology	List of specific parameters
LoRaWAN	• SF $\in [7; 12]$ • Coding rate $\in [1; 4]$ • CRC $\in \{0, 1\}$ • Type of traffic $\in \{unconfirmed, confirmed\}$
5G mmWave	• Numerology $\in [1; 5]$ • Hybrid Automatic Request $\in \{0, 1\}$ • Acknowledged mode (RLC-AM) $\in \{0, 1\}$
Wi-Fi HaLow	• MCS $\in [0; 9]$ • Spatial streams $\in [1; 3]$
Wi-Fi	• MCS $\in [0; 9]$ • Spatial streams $\in [1; 3]$ • Packet aggregation $\in \{0, 1\}$
6LoWPAN	• Min. Backoff exponent $\in [0; 7]$ • Min. Backoff exponent $\in [3; 8]$ • Max. CSMA backoff $\in [0; 5]$ • Max. frame retries $\in [0; 7]$

Table 4: Specific parameters of some network technology models.

Application abstraction parts	Parameters	Value
End-devices	Minimal number Maximal number Battery capacity (Amperes.hour)	50 100 2.4
Workload	Traffic direction Message size (bytes) Minimal frequency (packets/second) Maximal frequency (packets/second)	Upstream 100 1 1
Environment	Type Scope (meters) Expected lifetime (days)	Indoor 100 730

Table 5: Example of a smart building application requirements.

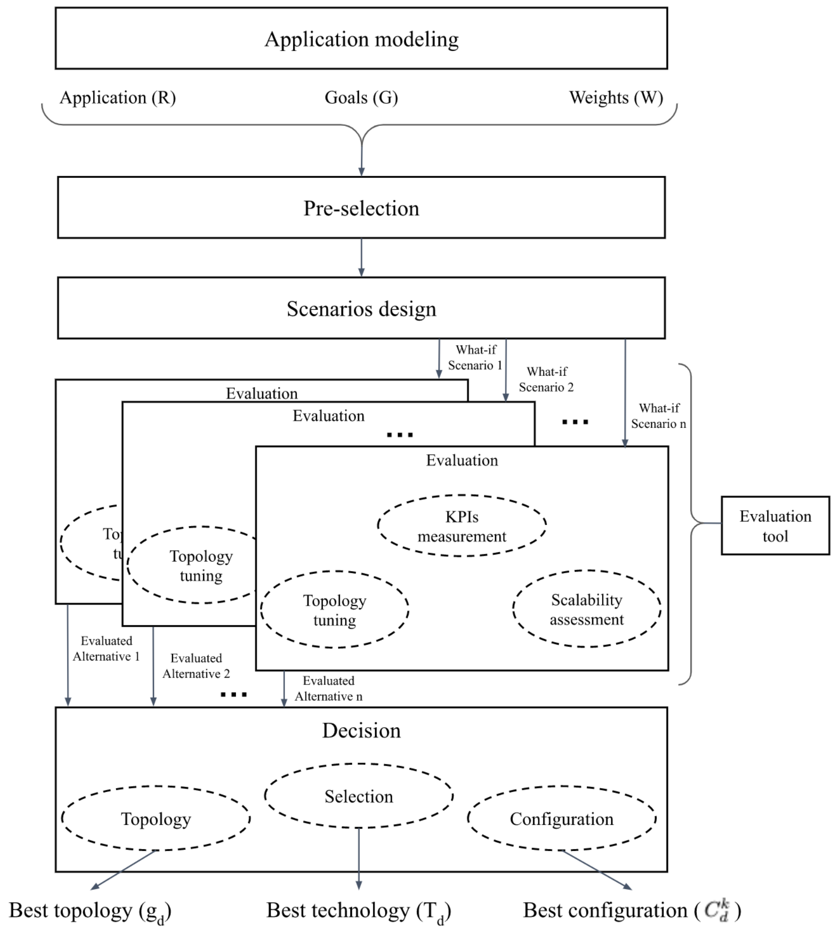

Figure 2: Overview of the HINTS steps.

4.3. Pre-selection

The goal of the **pre-selection** step is to dismiss network technologies that are obviously not meeting the application requirements. For example a network technology can be dismissed because its maximum data rate does not support the expected workload, derived from the message frequency and the message size. To do this, HINTS applies a filtering process, which can be implemented as a decision tree. The application requirements are compared to the maximum values of the message size and data rate that a network technology can provide. For instance, an application scenario with a traffic workload over 1 Mbps can never be satisfied with LoRaWAN. Then, there is no need for further analysis. The inappropriate network technologies are simply dismissed for the following steps.

KPI name	Unit	Goal	Weight
Message delivery rate	Percentage (%)	90	0.2
Battery lifetime	Days (d)	60	0.2
Energy consumption	Joules (J)	1	0.2
Message latency	Miliseconds (ms)	10	0.2
Network cost	Dollars ($)	-1	0.2

Table 6: Example of KPIs goals and weights.

4.4. Scenario design

After the **pre-selection** step, there is a need to explore in depth the network technologies candidates with various network configurations in order to compare them. The **scenario design** step consists in identifying the different network settings for the network technology candidates. Each setting represents a network technology candidate associated with network configuration parameters to be evaluated and compared to the others. This means that for a single network technology, there can be various network configurations where each one represents a what-if scenario (for instance, LoRaWAN with SF7 will be considered differently than LoRaWAN with SF12). Note that most of the considered network configuration parameters are naturally bounded (*e.g.*, SF, from 7 to 12).

Scalability and evolutivity assessment

Most IoT deployments are expected to evolve over time, for instance in terms of network density (number of end-devices) or in terms of traffic workload (message frequency and message size). The future behavior of a network technology under these conditions must also be evaluated. HINTS recommends to design scenarios with the maximum number of end-devices and the heaviest traffic workloads. To this end, every what-if scenario is composed of a minimal deployment (with the minimal number of end-devices and the minimal message frequency), and a maximal deployment (with the maximal number of end-devices and the maximal message frequency). Recall that these parameters have been defined in Section 3.2. This will provide insights about the scalability and evolutivity of the different what-if scenarios.

4.5. Evaluation

The **evaluation** step consists in instantiating the what-if scenarios defined at the **scenario design** step described above for calculating their respective KPI values, for both the minimal and the maximal deployments. For the evaluation, real experimentation or simulation tools can be used. If experimentation is used, end-devices and monitoring tools must be set up and activated to capture the traces. Then the traces have to be analyzed and the KPIs computed. For small scale projects and a limited number of technologies and scenarios, this can be done in labs and within reasonable time. If the number of scenarios or end-devices is important, the experimentation may be impossible to perform. Moreover, experimenting the variety of technologies requires rare talents in all these networks technologies. Simulation can be considered as a better approach to get decent evaluation of the KPIs. Even if simulation can be viewed as providing approximate results, it allows relative scalability and configurability, which are required to explore different configurations for each network technology.

Topological considerations

For a given network technology with its configuration, the number of gateways deployed, g, referred to as the topology throughout this paper, can greatly impact the application performance. Increasing the parameter g may have three main implications on the network behavior and the communications performance:

1. Reducing the workload per gateway: We assume that each gateway has a dedicated channel so that the workload at each gateway decreases proportionally with the total number of gateways. This assumption is realistic for many technologies that have multiple orthogonal channels (*e.g.*, 64 in 868 MHz for LoRaWAN, 24 in 5 GHz for Wi-Fi, etc.).

2. Reducing the maximum distance between end-devices and their associated gateway: In our application model, we consider that the maximum distance between an end-device and its associated gateway is:

$$d = D/(2 * g) \tag{1}$$

   where D is the deployment scope (the maximum distance between two end-devices, see Sec. 3.2). This simple relation reflects that, in general, the more gateways, the closer the end-devices are from their gateway.

3. Increasing the cost of the solution: It can incur additional costs in the purchase, but also in the deployment and the maintenance of the gateways.

Therefore, the ideal topology (ideal number of gateways has to be determined for each what-if scenario. The HINTS approach is to evaluate each what-if scenario with an increasing number of gateways, g, starting at the minimal number (typically 1). The parameter g is iteratively increased by 1 until either KPIs goals of all the threshold-based KPIs (namely, message delivery, battery lifetime and message latency) are reached (depending on the defined goals G), or the improvement on these (threshold-based) KPIs is below a given value ϵ. To include a safety margin, the upper bound on g is incremented by one. Overall, the number of explored topologies for each what-if scenario is simply equal to the maximal number of gateways that were iteratively tested. This process is described in Algorithm 2. Throughout the paper, we define an alternative as a what-if scenario associated with a topology.

HINTS proposes the following formula to compute the cost KPI, including the deployment (network modules of the end-devices, and gateways) and the maintenance costs (*i.e.,* battery change):

$$Cost = \underbrace{p_{gw} * n_{gw} + p_{ed} * n_{ed}}_{Deployment} + \underbrace{(l/b) * p_{br} * n_{ed}}_{Maintenance} \tag{2}$$

where each parameter is defined in Table 7. The ratio of l (expected application lifetime) on b is used to calculate the number of times the end-devices' batteries will have to be replaced. Note that b (battery lifetime) is the only KPI whose value is derived from the simulation and not obtained directly.

p_{gw}	Price of a gateway
n_{gw}	Number of gateways
p_{ed}	Price of a network module for the end-device
n_{ed}	Number of end-devices
l	Expected scenario lifetime
b	Battery lifetime
p_{br}	Cost of a battery replacement

Table 7: Cost function parameters.

Algorithm 2 Evaluation process

Inputs:
$R = [R_1, \ldots, R_k]$; Application requirements;
$T = [T_1, \ldots, T_m]$; Pre-selected IoT network technologies;
$C = [C_1, \ldots, C_m]$; Network configurations;
$G = [G_1, \ldots, G_n]$, $G_i \in \mathbb{R}$; The KPIs performance goals (either a KPI threshold, or $G_i < 0$ means that the i-th KPI is without threshold);
$\epsilon \in \mathbb{R}$; Minimum improvement for KPIs;
Variables:
$q \in \mathbb{N}$; Number of evaluated alternatives;
$P = (p_{ij}) \in \mathbb{R}^{q \times 2n}$; KPI values of the alternatives;
Algorithm:
$ind = 1$;
for each i in $[1;m]$ **do**
  $g \leftarrow 1$; $limit[i] \leftarrow \infty$; $search \leftarrow True$;
  $p_0 \leftarrow \underbrace{[0, \ldots, 0]}_{n}$
  while $g \leq limit[i]$ **do**
    for each j in $[1;|C^i|]$ **do**
      $p_{ind} \leftarrow$ Evaluation $(R,\ C_i^j,\ g)$ // Evaluation returns the KPI values for the network configuration C_i^j, with g gateways.
      if (KPIs_Satisfied (p_{ind}, G) or Improvement $(p_{ind}, p_{ind-1}) \leq \epsilon$ and $search = True$ **then**
        $limit[i] \leftarrow g + 1$
        $search \leftarrow False$
      end if
      $ind \leftarrow ind + 1$;
    end for
    $g \leftarrow g + 1$
  end while
end for
$q \leftarrow ind$

4.6. Decision

The goal of the **decision** step is to compare and rank the alternatives evaluated in the **evaluation** step. The KPI values obtained in the **evaluation** step are stored in a matrix P. Depending on the class of the KPI (see Section 3.3), its original value is kept or it is caped (or floored) by a threshold, as shown in Algorithm 3.

For the second class (threshold-based KPIs), in the case of a minimum threshold (*e.g.*, for message delivery or battery lifetime), we have:

$$f(x, \alpha) = \begin{cases} x & \text{if } x > \alpha \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Algorithm 3 Applying filters to KPIs

Inputs:
$q \in \mathbb{N}$; Number of evaluated alternatives;
$G = [G_1, \ldots, G_n]$, $G_i \in \mathbb{R}$; The KPI performance goals ($G_i < 0$ means that the i-th KPI is without threshold);
$P = (p_{ij}) \in \mathbb{R}^{q \times 2n}$; KPI values of the alternatives;
Algorithm:
/* Applying KPI thresholds */
for each i in $[1, q]$ **do**
  for each j in $[1, 2n]$ **do**
    if $G_j \geq 0$ **then**
      $p_{ij} \leftarrow f(p_{ij}, G_j)$ // KPI with threshold (see Eqs. 3 & 4)
    end if
  end for
end for

and for a maximum threshold (*e.g.*, message latency), we have:

$$f(x, \alpha) = \begin{cases} \alpha & \text{if } x < \alpha \\ x & \text{otherwise} \end{cases} \tag{4}$$

where x denotes a KPI and α its associated threshold.

Then, the KPIs values are normalized as shown in Algorithm 4:

Algorithm 4 Normalization process

Inputs:
$P = (p_{ij}) \in \mathbb{R}^{q \times 2n}$; KPI values of the alternatives;
Variables:
$N = (n_{ij}) \in \mathbb{R}^{q \times 2n}$; Normalized KPIs;
Algorithm:
/* Normalization */
for each i in $[1, q]$ **do**
  for each j in $[1, 2n]$ **do**
    $n_{ij} \leftarrow \dfrac{p_{ij}}{\sqrt{\sum_{i=1}^{q} (p_{ij})^2}}$
  end for
end for

The results are ranked according to a score, obtained through a method derived from the TOPSIS MADM algorithm [27]. In HINTS, the ranking leverages (i) KPIs weights and (ii) a scalability factor set by IoT architects, on the basis on their knowledge of the business context. The KPIs weighting is done using a vector of preference, more commonly named weights, in form of $W = [W_1, \ldots, W_n]$ where $W_j \in \mathbb{R}, \sum_{j=1}^{n} W_j = 1$. The scalability factor, $\beta \in \{0, 1, 2\}$, determines which one of the minimal or the maximal deployment has more importance for decision making ($\beta = 0$ or $\beta = 2$, respectively), or if they have the same importance ($\beta = 1$) according to the IoT architect. If the scalability of the solution in terms of number of end-devices or in workload intensity (maximal number of end-devices and maximal message frequency, respectively) is critical, the architect will give a high value, namely 2, to the scalability factor. The weighted KPIs will then be multiplied by this scalability factor for the "at scale" (aka the maximal deployment) evaluation of a scenario. This process is detailed in Algorithm 5. Note that, for simplicity purposes, we consider that the obtained KPIs of alternatives are organized as follows: The first n KPI values correspond to the minimal deployment, whereas the n remaining KPI values correspond to the maximal deployment, as shown in Equation 5.

$$p_i = [\underbrace{p_{i1}, \ldots, p_{in},}_{Min.deployment} \underbrace{p_{i(n+1)}, \ldots, p_{i2n}}_{Max.deployment}] \tag{5}$$

Algorithm 5 **Weighting process**

Inputs:
$W = [W_1, \ldots, W_n]$, $W_i \in \mathbb{R}_+$, $\sum_{i=1}^{n} W_i = 1$; KPIs weights;
$N = (n_{ij}) \in \mathbb{R}^{q \times 2n}$; Normalized KPIs;
$\beta \in \{0, 1, 2\}$; Scalability factor;
Variables:
$V = (\mathrm{v}_{ij}) \in \mathbb{R}^{q \times 2n}$; Weighted normalized KPIs;
Algorithm:
/* Weighting */
for each i in $[1, q]$ **do**
  for each j in $[1, 2n]$ **do**
    $v_{ij} \leftarrow W_j \times n_{ij}$
  end for
  /* Apply the scalability factor to the KPIs obtained for the maximal deployment*/
  for each j in $[n + 1, 2n]$ **do**
    $v_{ij} \leftarrow \beta \times v_{ij}$
  end for
end for

HINTS calculates the positive ideal solution (best one) and the negative ideal solution (worst one) based on the range of estimated KPIs values. Then, a score is given to each alternative depending on the Euclidean distances between the considered alternative and the positive and negative ideal solutions. The way of calculating the positive and the negative ideal solutions as well as the scores is described in Algorithm 6.

Algorithm 6 Ranking process

Inputs: $V = (\mathrm{v}_{ij}) \in \mathbb{R}^{q \times 2n}$; Weighted normalized KPIs
Variables:
$V^+ = [v_1^+, \ldots, v_{2n}^+]$, $v_i^+ \in \mathbb{R}$; Ideal positive solution;
$V^- = [v_1^-, \ldots, v_{2n}^-]$, $v_i^- \in \mathbb{R}$; Ideal negative solution;
$S^+ = [s_1^+, \ldots, s_q^+]$, $s_i^+ \in \mathbb{R}$; Positive distances;
$S^- = [s_1^-, \ldots, s_q^-]$, $s_i^- \in \mathbb{R}$; Negative distances;
Algorithm:
/* Ranking */
for each j in $[1, 2n]$ **do**
  $v_j^+ \leftarrow Argmax\{v_{ij}, i = 1, \ldots, q\}$
  $v_j^- \leftarrow Argmin\{v_{ij}, i = 1, \ldots, q\}$
end for
for each i in $[1, q]$ **do**
  $s_i^+ \leftarrow \sqrt{\sum_{j=1}^{2n} (v_j^+ - v_j^-)^2}$
  $s_i^- \leftarrow \sqrt{\sum_{j=1}^{2n} (v_j^- - v_j^-)^2}$
end for
for each i in $[1, q]$ **do**
  $S_i \leftarrow \dfrac{s_i^-}{s_i^- + s_i^+}$
end for

Finally, the output of the **decision** step is the alternative that obtains the highest score, according to this ranking.

4.7. Summary of the methodology

Thanks to HINTS, the IoT architect will be able to leverage the network knowledge previously encoded by network experts, and make wise decisions by:

1. Quantifying the application requirements, identifying the KPIs performance goals and weighting them in the **application modeling** step.

2. Quantifying the KPIs performance goals to allow the assessment of the network configuration to the specific application context and its potential scale in the future in terms of number of devices as well as message frequency, still in the **application modeling** step.

3. Dismissing network technologies that are obviously inappropriate (do not meet the application requirements) via the **pre-selection** step.

4. Specifying detailed what-if scenarios (with network configurations) for the remaining network technologies candidates for in-depth performance and scalability analysis with the **scenario design** step.

5. Evaluating these what-if scenarios with different topologies for the minimal and the maximal number of end-devices and message frequency, in the **evaluation** step.

6. Comparing the alternatives (with network configurations and topologies) and selecting the best one with the **decision** step.

5. Case studies

In this section, we illustrate the implementation of HINTS methodology and its application on three case studies derived from real-life examples: Smart building, event video-surveillance and precision agriculture. We illustrate how HINTS can be leveraged to support the following decisions and context:

- Case A: Network technology and topology decision at the design phase of a tailored smart building solution with a potentially growing number of end-devices.

- Case B: Network technology and topology decision for the design phase of a pre-packaged event video-surveillance solution with a potentially growing traffic workload.

- Case C: Network configuration decision at the deployment phase of a pre-packaged LoRaWAN-based precision agriculture solution.

The HINTS methodology implementation tool [42] provides the following set of network technologies: (i) LoRaWAN, (ii) Wi-Fi HaLow (aka IEEE 802.11ah on the 868 MHz frequency band), (iii) Wi-Fi (namely, IEEE 802.11ac on the 5 GHz frequency band), (iv) 6LoWPAN and (v) Private 5G (mmWave on the 28 GHz frequency band). HINTS defines their network configuration parameters as the ones specified in Table 4.

In the HINTS implementation, the **pre-selection** step is based on the maximum data rate and the maximum message size for each considered network technology. Table 8 enumerates the different "theoretical" values proposed by HINTS for these parameters.

	LoRaWAN [45]	Wi-Fi [46]	Wi-Fi [43] HaLow	6LoWPAN [47]	5G [44] mmWave
Maximum data rate	50 Kbps	3.4 Gbps	234 Mbps	250 Kbps	10 Gbps
Maximum message size	256 B	65535 B	65535 B	65535 B	65535 B

Table 8: Numerical values for the maximal data rate and message size on the subset of network technologies, considered in the HINTS implementation tool.

For the **evaluation** step, HINTS implementation uses the release 3.33 of ns-3 for Wi-Fi and 6LoWPAN and resorts to code patches not integrated in the official version of ns-3 but widely used by the research community to evaluate LoRaWAN [48] and Wi-Fi HaLow [49], and the module developed in [50] for 5G mmWave. The length of simulations is determined so that there are at least 200 packets sent per end-device.

The improvement value, ϵ, to determine the ideal topology is set to 5% (see Section 4.5). The prices of end-devices (ED) and gateways (GW) used to compute the cost of an alternative are reported in Table 9. The maintenance cost corresponds to the battery replacement. In HINTS, the price of a battery replacement (parameter p_{bc} in Equation 2 and Table 9) depends on the application scenario environment. It is set to 5 USD for indoor and urban environments, and 50 USD for rural environment (see Table 7). Since the considered network technologies operate on unlicensed frequency bands or private environments, there are no additional band subscription fees.

Network technology	ED Price (USD)	GW Price (USD)
LoRaWAN [14]	5	1000
Wi-Fi HaLow [51]	15	1000
Wi-Fi [52]	10	100
6LoWPAN	30	200
5G mmWave	20	500

Table 9: Network equipment price for some network technologies considered for our case studies.

For all our case studies, the end-devices are expected to run on batteries. Therefore we keep only the battery lifetime KPI as it is correlated to the energy consumption.

In the **decision** step, and for the sake of simplicity, uniform weights are used for every KPI, and a scalability factor of 1 is used as well, so that the initial and the maximal deployments have the same importance.

5.1. Case A: Network technology and topology decision for a smart building solution

This case study is devoted to the design of a tailored smart building solution, where sensors will collect periodical measurements (room temperature and occupancy sensors, air quality, etc.) to maintain safety and comfort within the facility. The structure of the building is the following: We consider 20, 10 and 50 meters for its length, width and height, respectively, with 16 floors of 6 rooms in each floor.

In the **application modeling** step of HINTS, the application scenario is defined as follows: We consider 50 and 100 sensors for the minimal and the maximal number of end-devices, respectively, equipped with 2,400 mAh capacity batteries (powered by 3 V). The sensors send 1 packet of 100 bytes every second to their gateway. The maximal message frequency is equal to 1 message per second as well. The environment is indoor, since the application operates inside a building. The sensors are randomly placed inside the building around a gateway, with a deployment scope of 100 meters (which is approximately equal to the maximum distance that could separate two points inside that building). For the KPIs goals, this application scenario would require batteries to last at least 3 months, a message latency below 100 ms and a message delivery above 90%. For the cost calculation, the parameter l (expected scenario lifetime, see Section 4.5) is set to 2 years.

At the **pre-selection** step, LoRaWAN is dismissed from the list of network technologies candidates since the message frequency required for this case study (1 packet per second) is too high for the maximum data rate of LoRaWAN (see Table 8). 5G mmWave is also dismissed since these frequency bands are not expected to be used in this kind of application scenarios, due to their poor penetration capacity [53]. Hence, the remaining network technologies are Wi-Fi, 6LoWPAN and Wi-Fi HaLow.

At the **scenario design** step, we consider the following network configurations for the remaining network technologies: For Wi-Fi, it is a channel width of 80 MHz, one spatial stream, a long guard interval and no frame aggregation. For Wi-Fi HaLow, it is a channel width of 2 MHz, a long guard interval, a beacon interval of 51200 ms and one RAW group. For 6LoWPAN, it is a channel width of 5 MHz, a number of frame retries of 4, a number of CSMA backoffs set to 5 and the maximum (resp. minimum) backoff exponent set to 4 (resp. 3). Note that these values are used as a default network configuration, and other parameters can be considered for further study. The same remark applies to the remaining case studies.

The simulation time is set to 200 seconds in the **evaluation** step. We present the results of the **evaluation** step in Table 10. In this step, HINTS iterates to determine the ideal topology. In this example, the KPIs performance goals are met for the threshold-based KPIs with one gateway for Wi-Fi HaLow and two gateways for 6LowPAN. For Wi-Fi, we see that the goals are attained for message delivery, message latency and battery lifetime with 4 gateways. Therefore, an additional study for Wi-Fi with 5 gateways is considered. We observe that the number of alternatives to consider differs for each network technology. We notice that, unlike 6LoWPAN and Wi-Fi, Wi-Fi HaLow manages to keep the message delivery at 100% with one gateway, for both the minimal number of end-devices of 50 and the maximal number of end-devices of 100. We also see that, despite being the most performing in terms of battery lifetime and cost, Wi-Fi HaLow is outperformed by Wi-Fi and 6LoWPAN in terms of message latency.

Table 10 shows that the **decision** step determines Wi-Fi HaLow with two gateways as being the best alternative among those considered. Figure 3 uses a radio chart to reflect the resulting KPIs after the application of the function f for the threshold-based KPIs (see Eqs. 3 and 4), the normalization and the weighting processes. Then, each resulting KPI value is divided by the maximum (for message delivery and battery lifetime) or minimum (for message latency or cost) value of that KPI among all the alternatives. These values are finally plotted in the radio chart. Note that for readability purposes, we display the cost efficiency and latency efficiency, which are the inverse values of cost and message latency, respectively. Also, the "Latency Efficiency" axis is displayed using a logarithmic scale. We see that Wi-Fi HaLow, regardless of the number of gateways, significantly outperforms the other alternatives in terms of battery lifetime KPI. According to the calculated scores, the **decision** step returns HaLow with 2 gateways as the network technology and topology to opt for the design phase of this application scenario.

Network technology		Minimal deployment (50 end-devices)				Maximal deployment (100 end-devices) *Scalability factor: 1*				
Technology	Nb. of GW	Message delivery *Weight: 1* *Unit: %* *Goal: >90*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >80*	Message latency *Weight: 1* *Unit: ms* *Goal: <100*	Cost *Weight: 1* *Unit: $*	Message delivery *Weight: 1* *Unit: %* *Goal: >90*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >80*	Message latency *Weight: 1* *Unit: ms* *Goal: <100*	Cost *Weight: 1* *Unit: $*	Score
Wi-Fi	1	42.0	61.72	0.05	3850	30.0	49.1	0.05	9100	0.02
Wi-Fi	2	80.0	66.28	0.05	3700	86.0	61.24	0.05	7700	0.07
Wi-Fi	3	87.5	66.45	0.05	3800	96.97	85.86	0.05	5800	0.32
Wi-Fi	4	100.0	89.09	0.05	3150	100.0	88.38	0.05	5900	0.46
Wi-Fi	5	100.0	89.27	0.05	3150	100.0	88.71	0.05	5900	0.46
HaLow	1	100.0	362.16	48.41	2250	100.0	277.78	57.28	3500	0.87
HaLow	**2**	**100.0**	**421.69**	**48.9**	**3000**	**100.0**	**331.8**	**58.72**	**4500**	**0.93**
6LoWPAN	1	54.31	91.76	29.62	3700	44.63	71.44	12.61	9700	0.12
6LoWPAN	2	94.46	125.07	12.38	3400	88.29	85.75	21.67	7400	0.36
6LoWPAN	3	98.09	142.95	16.47	3350	94.10	112.28	7.46	7100	0.49

Table 10: Case A: smart building results.

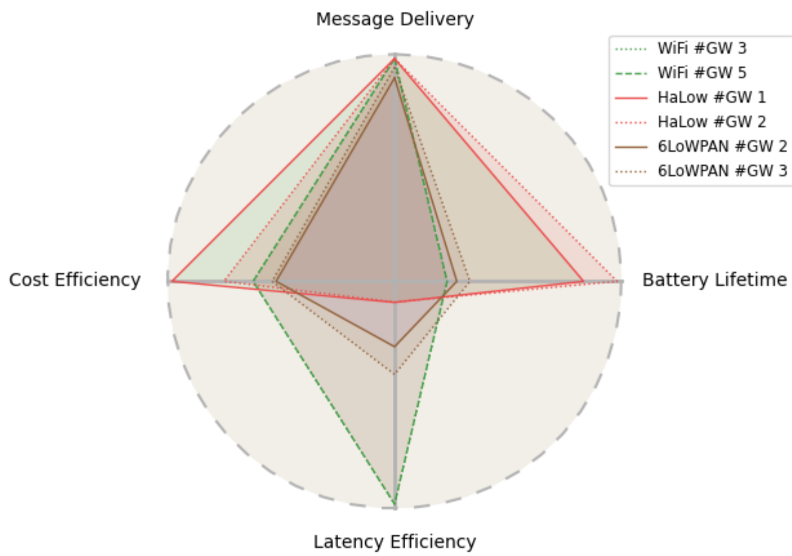

Figure 3: Case A: smart building radio-chart.

5.2. Case B: Network technology and topology selection for an event video-surveillance application

We consider the design of a pre-packaged camera-based surveillance solution for events gathering large crowds (*e.g.*, trade fair, concert, etc.). Short installation time and stringent logistic constraints impose wireless connectivity with self-powered cameras placed at specific locations.

In the **application modeling** step, the application scenario is defined as follows: This solution is sold for an minimal (and maximal) number of 30 cameras, each one equipped with 2,400 mAh capacity batteries (powered by 3 V). The cameras typically send 200 packets of 1500 bytes per second to their gateway (leading to a workload of 2 Mbps). The application developers of the solution want to be able to improve the precision of the images. At maximal traffic workload, the cameras will be sending 300 packets every second to the gateway. This will result in a higher bandwidth requirement: A workload of 3 Mbps. The cameras are randomly placed around a gateway and the deployment scope is about 60 meters, in an outdoor urban environment. For the KPIs threshold, the architect specifies a message delivery above 95%, batteries to last at least a week, and a message latency under 10 ms. The parameter l (expected lifetime) is set to two months, including installation and exploitation phases.

Due to the large expected workload (2 and 3 Mbps), the **pre-selection** step dismisses LoRaWAN, Wi-Fi HaLow and 6LoWPAN from the possible network technologies candidates. Thus, only Wi-Fi and 5G mmWave remain.

Regarding the **scenario design** step, the existing Wi-Fi configuration is the same as for case A, whereas 5G mmWave uses a 5G NR numerology of 2, disabled HARQ and RLC-AM.

At the **evaluation** step, the traffic workload defined in the **application modeling** step leads to a simulation time of around 2.5 seconds. HINTS iterates on the number of gateways. In this example, two gateways are enough for Wi-Fi to reach the KPIs goals of the threshold-based KPIs. Thus, we consider three alternatives for Wi-Fi with a number of gateways ranging from one to three. Regarding 5G mmWave, the improvement obtained when augmenting the number of gateways from 1 to 2 does not exceed ϵ, therefore three gateways are sufficient for the study (taking into account the safety margin). We present the results of the **evaluation** step in Table 11. First, we see that the message delivery attains 100.0% starting from Wi-Fi with two gateways, while its value is around 55% for one gateway, which means that one gateway is not enough to support the whole traffic workload. It attains practically 100 % with one gateway for 5G mmWave. Also, we notice a slight increase in battery lifetime when the number of gateways is increased. Indeed, the more gateways, the less contention, resulting in end-devices spending more time in an idle state, which consumes less energy. The same remark cannot be made for 5G mmWave: The battery lifetime is not impacted by the number of gateways. Moreover, the obtained battery lifetime does not even last 1 day, while it manages to go up to 13 days for Wi-Fi with two gateways. The high energy consumption in 5G mmWave seems in line with the work [54] in which the authors showed that the 5G mmWave has substantial energy and computing power. In the same way, message latency slightly decreases for the same reason (less contention) for both network technologies. Regarding the cost, it tends to get lower with the number of gateways for Wi-Fi, which is due to the associated decreasing number of battery replacements, contrarily to 5G mmWave, where the cost seems to increase.

Overall, Table 11 shows that the Wi-Fi alternative with 3 gateways obtains the best score and outperforms the others. The trade-offs between the different KPIs are captured in Figure 4 (computed as in Section 5.2) for the two best alternatives of each network technology. Figure 4 clearly shows that the Wi-Fi alternative with 3 gateways outperforms the other alternatives in terms of battery lifetime and cost efficiency.

Network technology		Minimal deployment (2 Mbps)				Maximal deployment (3 Mbps) *Scalability factor: 1*				
Technology	Nb. of GW	Message delivery *Weight: 1* *Unit: %* *Goal: >95*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >7*	Message latency *Weight: 1* *Unit: ms* *Goal: <10*	Cost *Weight: 1* *Unit: $*	Message delivery *Weight: 1* *Unit: %* *Goal: >95*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >7*	Message latency *Weight: 1* *Unit: ms* *Goal: <10*	Cost *Weight: 1* *Unit: $*	Score
Wi-Fi	1	55.85	4.81	1.53	1050	22.23	4.21	2.86	1200	0.43
Wi-Fi	2	100.0	13.57	0.2	1100	99.99	9.7	0.22	1400	0.82
Wi-Fi	**3**	**100.0**	**17.37**	**0.21**	**1050**	**100.0**	**15.61**	**0.25**	**1200**	**0.99**
5G mmWave	1	99.96	0.14	1.01	64100	99.97	0.14	1.05	64100	0.35
5G mmWave	2	100	0.1	0.95	90100	100	0.1	0.97	90100	0.32
5G mmWave	3	100	0.13	1.0	68400	100	0.13	1.0	68400	0.34

Table 11: Case B: event video-surveillance results.

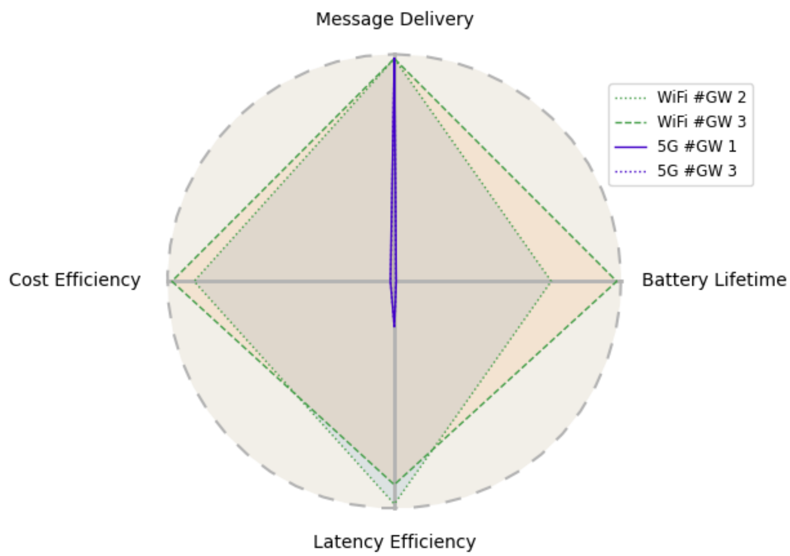

Figure 4: Case B: event video-surveillance radio-chart.

5.3. Case C: Network configuration decision for a precision agriculture application

In this case study, we consider the deployment of a pre-packaged precision agriculture solution in a given farm, using LoRaWAN. The solution architect needs to adjust the network configuration parameters of this solution taking into account the specificity of the deployment. The precision agriculture system comprises humidity, temperature and PH sensors, which measure these metrics before sending them to a LoRaWAN gateway for further transmission and processing.

At the **application modeling** step, the application scenario is defined as follows: The minimal number of sensors is 200, whereas the maximal number of sensors is 300, each one equipped with the same batteries as for Case A and B (2,400 mAh powered by 3 V). The sensors send one packet of 30 bytes every 3 minutes to their gateway, in an outdoor rural environment. The sensors are placed around the gateway and the deployment scope is 3,000 meters. Regarding the KPIs, this solution deployment typically requires a battery lifetime of 1 year, a message latency lower than 1 second and a message delivery of at most 90%. The value of l is set to 10 years.

Since the network technology to be used has already been decided (LoRaWAN), the **pre-selection** step is skipped.

In the **scenario design** step, the goal is to explore the various network configurations for the LoRaWAN settings within the end-devices. The goal is to determine which SF to select as well as which coding rate and type of traffic (confirmed or unconfirmed) to use. Several network configurations are generated accordingly. We consider the minimal and the maximal values for each parameter.

In the **evaluation** step, the traffic workload leads to 36,000 seconds of simulation time to ensure a minimum of 200 packets sent by each sensor. This pre-packaged solution supports a single gateway. Table 12 presents the KPI values of the various LoRaWAN alternatives. First, we see that the SF has a tremendous impact on the KPIs, where the less the better: SF7 allows to ameliorate the performances almost by a factor of 2. Then, the type of traffic strongly influences the battery lifetime: In case the traffic is unconfirmed, it is around 8 times higher than with unconfirmed traffic. This is due to the re-transmissions triggered following up the loss of a packed when the traffic is confirmed.

In Table 12, the **decision** step elects LoRaWAN with a SF7, a 1 coding rate, with an unconfirmed traffic as the best alternative. The differences between the best alternatives' performances are shown in Figure 5 (computed as in Section 5.2).

Network configuration			Minimal deployment (200 end-devices)				Maximal deployment (300 end-devices) *Scalability factor: 1*				
SF	Coding rate	Traffic type	Message delivery *Weight: 1* *Unit: %* *Goal: >90*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >730*	Message latency *Weight: 1* *Unit: ms* *Goal: <1000*	Cost *Weight: 1* *Unit: $*	Message delivery *Weight: 1* *Unit: %* *Goal: >90*	Battery lifetime *Weight: 1* *Unit: d* *Goal: >730*	Message latency *Weight: 1* *Unit: ms* *Goal: <1000*	Cost *Weight: 1* *Unit: $*	Score
7	**1**	**0**	**95.78**	**2560.7**	**82.17**	**12000**	**95.73**	**2560.7**	**82.17**	**17500**	**0.98**
7	1	1	99.44	331.37	82.17	112000	99.19	332.6	82.176	167500	0.51
7	4	0	93.65	1819.54	82.17	22000	93.74	1819.54	82.176	17500	0.81
7	4	1	98.65	232.95	82.17	152000	97.83	234.18	82.176	227500	0.5
12	1	1	43.47	126.44	197.4	282000	31.97	114.1	197.4	467500	0.0
12	1	0	43.31	126.44	197.4	282000	31.78	112.7	197.4	482500	0.38
12	4	1	33.29	126.77	197.4	282000	22.33	232.95	197.4	422500	0.38
12	4	1	33.11	126.77	197.4	282000	22.13	232.95	197.4	422500	0.38

Table 12: Case C: precision agriculture results.

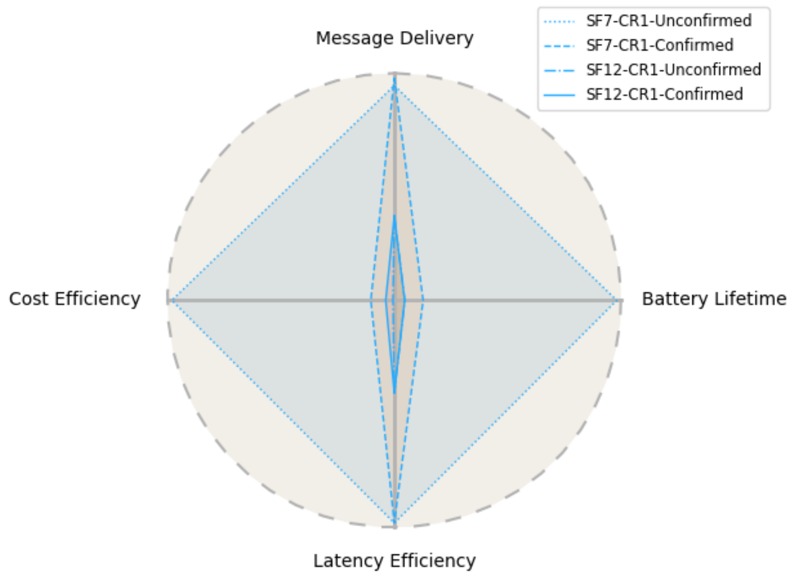

Figure 5: Case C: precision agriculture radio-chart.

Table 13 provides a summary of the **application modeling** for the case studies A, B and C. An outline of the rest of the steps is given in Table 14.

Application modeling	Parameters	Cases		
		A	B	C
End-devices	• Minimal number • Maximal number • Battery capacity (Amperes.hour)	50 100 2.4	30 30 2.4	200 300 2.4
Workload	• Traffic direction • Message size (bytes) • Minimal frequency (packets/second) • Maximal frequency (packets/second)	Upstream 100 1 1	Upstream 1500 200 300	Upstream 30 0.005 0.005
Environment	• Type • Scope (meters) • Expected lifetime (days)	Indoor 50 730	Urban 30 60	Rural 1500 3650

Table 13: Application modeling of cases A, B and C.

Step / Case	Application modeling		Pre-selection	Scenario design	Evaluation	Decision
	Goals	Weights				
A	Delivery $>$ 90% Lifetime $>$ 80 days Latency $<$ 100 ms	Delivery: 0.25 Lifetime: 0.25 Latency: 0.25 Cost: 0.25	Wi-Fi Wi-Fi HaLow 6LoWPAN	Single network configuration	See Table 10	Wi-Fi HaLoW with 1 GW
B	Delivery $>$ 95% Lifetime $>$ 7 days Latency $<$ 10 ms	Delivery: 0.25 Lifetime: 0.25 Latency: 0.25 Cost: 0.25	Wi-Fi 5G mmWave	Single network configuration	See Table 11	Wi-Fi with 4 GW
C	Delivery $>$ 90% Lifetime $>$ 365 days Latency $<$ 1000 ms	Delivery: 0.25 Lifetime: 0.25 Latency: 0.25 Cost: 0.25	LoRaWAN	SF $\in \{7, 12\}$ Coding rate $\in \{1, 4\}$ Traffic type $\in$ {Unconfirmed (0), Confirmed}	See Table 12	SF=7 Coding rate=1 Traffic type= Unconfirmed

Table 14: Case Studies Summary.

5.4. Discussion

The computational complexity of HINTS implementation mostly resides in the **evaluation** step, because running simulations (for example in ns-3) can be time-consuming depending on the density of the simulated network and the simulation time. The **decision** step includes some relatively lightweight computations relating to the normalization, the weighting and the computing of the Euclidean distances (which results in a complexity of $O(n \times q)$, with q being the number of evaluated alternatives and n the number of KPIs). Overall, the **decision** step time is considered negligible in comparison to the evaluation time.

The availability of the simulator of a given network technology is a non-negligible constraint. In this paper, we demonstrate that such comparative simulations can bring a lot of insights rapidly and at a low cost to encourage network experts and researchers to contribute to the effort by coding open-source simulators and their HINTS models.

A current limitation of HINTS is the fact that it only considers star topologies for networks. HINTS could be further extended to include mesh topologies. A user-friendly interface may also help disseminate HINTS among IoT architects.

As for the issue of ranking reversal that may emerge in MADM methods, it does not apply in our case. Indeed, this classical problem refers to a change in the ordering among the alternatives, after the addition or the removal of an alternative from the group previously defined. For example, in the case of dynamic selection of a wireless interface, this alteration can affect the routing of packets [55]. Since HINTS targets static selection and gives recommendations prior to the effective network deployment, it is not subject to the ranking reversal problem. However, if we extend HINTS to address the emerging problem of dynamic reconfiguration of end-devices with multiple IoT networks, then we will have to deal with this issue.

To end this discussion, we compare HINTS with the work of [19] which is the closest to our solution. Both methodologies help in modeling the network technologies and the IoT application. [19] provides questionnaires to help IoT architects eliminate technology candidates based on deployment constraints and on the evaluation of a single KPI, namely financial cost, for decision making. On the other hand, HINTS relies on technical specification to eliminate network technology candidates, but then resorts to an automated evaluation of KPIs (including the financial cost and network performance metrics) resulting from discrete-event simulation and decision algorithms to select the "right" network technology, its configuration and topology. We believe these two solutions (HINTS and [19]) are complementary and could be nicely combined in future works.

6. Conclusion

In this paper, we have presented HINTS, a methodology for supporting IoT network technologies selection and configuration. HINTS relies on the modeling of IoT network technologies on one side and a five steps decision process on the other side. We have described the different steps of this process, which includes: (i) **Application modeling**, to abstract the IoT application specificity, its requirements on a set of KPIs; (ii) **pre-selection**, which dismisses the inappropriate network technologies; (iii) **scenario design**, to configure the application with the appropriate network technologies candidates, (iv) **evaluation** where the best suited topology is iteratively found and an instrument such as simulation is used to estimate the KPIs on the targeted application scenario for each alternative and (v) **decision**, which assigns scores to each alternative using a MADM method, derived from TOPSIS. We have presented three case studies inspired by real-life deployments to illustrate the application of HINTS. The results have shown that HINTS enables a fair and insightful comparison of IoT network technologies for a given application scenario. Moreover, it permits to explore and determine network configuration parameters and the number of gateways to deploy. This work highlights the importance of the application context, of the environment, and of the scaling factor in the network selection process and expected performance. For the sake of reproducibility, we made the source code available at [42].

In future works, we plan to extend HINTS to include IoT networks operating on mesh topology as well as the potential mobility of end-devices so that HINTS can handle connected vehicle applications or drone-based use-cases. We also aim at enriching the simulation models with real measurements derived from experiments conducted on testbeds such as FIT-IoT lab [20] and SLICES [56]. Adding more network technologies (NB-IoT, LTE-M, etc.) in HINTS and specially in the simulator (viz ns-3) is another item on our agenda. Finally, we plan to make the methodology available as an online service to assist IoT architects and network specialists in their everyday life, through an online no-code tool that we have initiated in [57].

7. Acknowledgment

This work was performed within the framework of the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR) and the support of Stackeo (https://stackeo.io). The authors wish to thank the anonymous referees for their thorough and constructive review of an earlier version of this paper.

Appendix A. Energy consumption models

Table A.15 indicates the numericals values we used throughout this paper to compute the energy consumption of the Wi-Fi, LoRaWAN, Wi-Fi HaLow and 6LoWPAN network interface cards (NICs). The values for Wi-Fi were selected calibrating the state machine against the measurements provided by Serrano et al. in [58]. The values for LoRaWAN are those given by default in the ns-3 module for the LoRaWAN consumption by Magrin et al. [48]. For 5G mmWave, the values are derived from [59].

State	LoRaWAN	5G mmWave	Wi-Fi HaLow	6LoWPAN	Wi-Fi
Tx	77	350	7.2	7	107
Rx	28	350	4.4	1.5	40
Idle	1	/	1	/	1
Sleep	0.015	45	/	/	/
CCA Busy	/	1	/	/	/
Switch	/	/	/	0.5	/

Table A.15: Drawn current values for each state of the machine state used in ns-3 simulations to evaluate the power consumption of LoRaWAN, 5G mmWave, Wi-Fi HaLow, Wi-Fi and 6LoWPAN communications.

Appendix B. Notation Table

Table B.16 details the used symbols used throughout the paper.

Symbol	Meaning
R	Application requirements
K	KPIs
G	KPIs performance goals
W	KPIs weights
T	Network technologies candidates
C	Network configuration parameters
P	KPIs values of the alternatives
N	Normalized KPIs
W	KPIs weights
V	Weighted normalized KPIs
V^+	Ideal positive solutions
V^-	Ideal negative solutions
S^+	Positive distances
S^-	Negative distances
q	Number of evaluated alternatives
ϵ	Minimum improvement for the KPIs
n	Number of KPIs
β	Scalability factor

Table B.16: Table of notation.

References

[1] M. Chui, M. Collins, and M. Patel, "The internet of things: Catching up to an accelerating opportunity," McKinsey Global Institute research report, 2021.

[2] ABI research, "Iot asset tracking device evolution: Technologies and features driving the next stage of growth," ABI research, 2020.

[3] A. Bassi, M. Bauer, M. Fiedler, T. Kramp, R. Van Kranenburg, S. Lange, and S. Meissner, Enabling things to talk. Springer Nature, 2013.

[4] L. F. Rahman, T. Ozcelebi, and J. Lukkien, "Understanding iot systems: a life cycle approach," Procedia computer science, vol. 130, pp. 1057–1062, 2018.

[5] D. Slama, IgniteWorx: Design and evaluation of a system-supported methodology for IIoT project setup. Technische Universitaet Berlin (Germany), 2019.

[6] Beecham Research, "Why IoT projects fail & how to beat the odds An executive summary," 2020.

[7] Microsoft, "IoT Signals - Manufacturing Spotlight," 2022.

[8] A. Aramayo-Prudencio et al., "Digital-manufacturing-escaping-pilot-purgatory," McKinsey Global Institute research report, 2018.

[9] W. Kassab and K. A. Darabkh, "A–z survey of internet of things: Architectures, protocols, applications, recent advances, future directions and recommendations," Journal of Network and Computer Applications, vol. 163, p. 102663, 2020.

[10] P. Kaňuch, D. Macko, and L. Hudec, "Survey: Classification of the iot technologies for better selection to real use," in 2020 43rd International Conference on Telecommunications and Signal Processing (TSP). IEEE, 2020, pp. 500–505.

[11] R. S. Sinha, Y. Wei, and S.-H. Hwang, "A survey on lpwa technology: Lora and nb-iot," Ict Express, vol. 3, no. 1, pp. 14–21, 2017.

[12] A. Ikpehai, B. Adebisi, K. M. Rabie, K. Anoh, R. E. Ande, M. Hammoudeh, H. Gacanin, and U. M. Mbanaso, "Low-power wide area network technologies for internet-of-things: A comparative review," IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2225–2240, 2018.

[13] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "Overview of cellular lpwan technologies for iot deployment: Sigfox, lorawan, and nb-iot," in 2018 ieee international conference on pervasive computing and communications workshops (percom workshops). IEEE, 2018, pp. 197–202.

[14] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of lpwan technologies for large-scale iot deployment," ICT express, vol. 5, no. 1, pp. 1–7, 2019.

[15] B. Vejlgaard, M. Lauridsen, H. Nguyen, I. Z. Kovács, P. Mogensen, and M. Sorensen, "Coverage and capacity analysis of sigfox, lora, gprs, and nb-iot," in 2017 IEEE 85th vehicular technology conference (VTC Spring). IEEE, 2017, pp. 1–5.

[16] Y. Lalle, L. C. Fourati, M. Fourati, and J. P. Barraca, "A comparative study of lorawan, sigfox, and nb-iot for smart water grid," in 2019 Global Information Infrastructure and Networking Symposium (GIIS). IEEE, 2019, pp. 1–6.

[17] Verhoeven, Richard et al., "Performance evaluation of wi-fi halow, nb-iot and lora for smart city applications," in PE-WASUN, 2022.

[18] S. Si-Mohammed, T. Begin, I. G. Lassous, and P. Vicat-Blanc, "Adiperf: A framework for application-driven iot network performance evaluation," in 2022 International Conference on Computer Communications and Networks (ICCCN). IEEE, 2022, pp. 1–8.

[19] F. Vannieuwenborg, S. Verbrugge, and D. Colle, "Choosing iot-connectivity? a guiding methodology based on functional characteristics and economic considerations," Transactions on emerging telecommunications technologies, vol. 29, no. 5, p. e3308, 2018.

[20] FIT IoT-Lab, iot-lab.info.

[21] Smart Santander, smartsantander.eu.

[22] Copelabs, copelabs.ulusofona.pt.

[23] Fiesta IoT, github.com/fiesta-iot.

[24] F. Bari and V. C. Leung, "Automated network selection in a heterogeneous wireless network environment," IEEE Network, vol. 21, no. 1, pp. 34–40, 2007.

[25] F. Bari and V. Leung, "Multi-attribute network selection by iterative TOPSIS for heterogeneous wireless access," 2007 4th Annual IEEE Consumer Communications and Networking Conference, CCNC 2007, pp. 808–812, 2007.

[26] M. A. Senouci, M. S. Mushtaq, S. Hoceini, and A. Mellouk, "TOPSIS-based dynamic approach for mobile network interface selection," Computer Networks, vol. 107, pp. 304–314, 2016.

[27] G.-H. Tzeng and J.-J. Huang, Multiple attribute decision making: methods and applications. CRC press, 2011.

[28] Gazis, Vangelis, et al., "Toward a generic" always best connected" capability in integrated wlan/umts cellular mobile networks (and beyond)," IEEE Wireless Communications, 2005.

[29] L. Wang and D. Binet, "Best permutation: a novel network selection scheme in heterogeneous wireless networks," in Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly, 2009.

[30] J. Hou and D. C. O'Brien, "Vertical handover-decision-making algorithm using fuzzy logic for the integrated radio-and-ow system," IEEE Transactions on Wireless Communications, 2006.

[31] W. Zhang, "Handover decision using fuzzy madm in heterogeneous networks," in IEEE Wireless Communications and Networking Conference (WCNC), 2004.

[32] Stevens-Navarro, Enrique, et al., "An mdp-based vertical handoff decision algorithm for heterogeneous wireless networks," IEEE Transactions on Vehicular Technology, 2008.

[33] Ning, Zhaolong, et al., "Markov-based vertical handoff decision algorithms in heterogeneous wireless networks," Computers & Electrical Engineering, 2014.

[34] M. Cesana, N. Gatti, and I. Malanchini, "Game theoretic analysis of wireless access network selection: models, inefficiency bounds, and algorithms," in Proceedings of the 3rd international conference on performance evaluation methodologies and tools, 2008.

[35] Liu, Bin, et al., "Ahp and game theory based approach for network selection in heterogeneous wireless networks," in IEEE 11th Consumer Communications and Networking Conference (CCNC), 2014.

[36] Y. Gai, B. Krishnamachari, and R. Jain, "Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations," IEEE/ACM Transactions on Networking, 2012.

[37] Henri, Sébastien, et al., "Multi-armed bandit in action: Optimizing performance in dynamic hybrid networks," IEEE/ACM Transactions on Networking, 2018.

[38] N. Rozanski and E. Woods, Software systems architecture: working with stakeholders using viewpoints and perspectives. Addison-Wesley, 2011.

[39] M. Stoffers and G. Riley, "Comparing the ns-3 propagation models," in 2012 IEEE 20th international symposium on modeling, analysis and simulation of computer and telecommunication systems. IEEE, 2012, pp. 61–67.

[40] R. K. Singh, P. P. Puluckul, R. Berkvens, and M. Weyn, "Energy consumption analysis of lpwan technologies and lifetime estimation for iot application," Sensors, vol. 20, no. 17, p. 4794, 2020.

[41] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J.-P. Vasseur, and R. Alexander, "Rpl: Ipv6 routing protocol for low-power and lossy networks," Tech. Rep., 2012.

[42] HINTS source code, https://github.com/SamirSim/Selection-Methodology-IoT.

[43] W. Sun, M. Choi, and S. Choi, "Ieee 802.11ah: A long range 802.11 wlan at sub 1 ghz," J. ICT Stand., vol. 1, pp. 83–108, 2013.

[44] A. Ahad, M. Tahir, M. A. Sheikh, K. I. Ahmed, A. Mughees, and A. Numani, "Technologies trend towards 5g network for smart health-care using iot: A review," Sensors (Switzerland), vol. 20, no. 14, pp. 1–22, 2020.

[45] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the Limits of LoRaWAN," IEEE Communications Magazine, vol. 55, no. 9, pp. 34–40, 2017.

[46] "Ieee standard for information technology–telecommunications and information exchange between systems—local and metropolitan area networks–specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications–amendment 4: Enhancements for very high throughput for operation in bands below 6 ghz." IEEE Std 802.11ac(TM)-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012), pp. 1–425, 2013.

[47] J. K. and A. Desai, "IoT: Networking Technologies and Research Challenges," International Journal of Computer Applications, vol. 154, no. 7, pp. 1–6, 2016.

[48] D. Magrin, M. Centenaro, and L. Vangelista, "Performance evaluation of lora networks in a smart city scenario," in 2017 IEEE International Conference on communications (ICC). ieee, 2017, pp. 1–7.

[49] L. Tian, S. Deronne, S. Latré, and J. Famaey, "Implementation and validation of an ieee 802.11 ah module for ns-3," in Proceedings of the Workshop on ns-3, 2016, pp. 49–56.

[50] M. Mezzavilla, M. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, "End-to-end simulation of 5g mmwave networks," IEEE Communications Surveys & Tutorials, vol. 20, no. 3, pp. 2237–2263, 2018.

[51] V. Baños-Gonzalez, M. S. Afaqui, E. Lopez-Aguilera, and E. Garcia-Villegas, "Ieee 802.11 ah: A technology to face the iot challenge," Sensors, vol. 16, no. 11, p. 1960, 2016.

[52] K. Pahlavan and P. Krishnamurthy, "Evolution and impact of wi-fi technology and applications: a historical perspective," International Journal of Wireless Information Networks, vol. 28, no. 1, pp. 3–19, 2021.

[53] B. T. Jijo, S. Zeebaree, R. R. Zebari, M. Sadeeq, A. B. Sallow, S. Mohsin, and Z. S. Ageed, "A comprehensive survey of 5g mm-wave technology design challenges," Asian Journal of Research in Computer Science, vol. 8, no. 1, pp. 1–20, 2021.

[54] M. H. Mazaheri, S. Ameli, A. Abedi, and O. Abari, "A millimeter wave network for billions of things," in Proceedings of the ACM Special Interest Group on Data Communication, 2019, pp. 174–186.

[55] B. Foubert and N. Mitton, "Rodent: a flexible topsis based routing protocol for multi-technology devices in wireless sensor networks," ITU Journal on Future and Evolving Technologies, vol. 2, no. 1, 2021.

[56] SLICES-RI, https://slices-ri.eu/.

[57] S. Si-Mohammed, M. Janumporn, T. Begin, I. Guérin Lassous, and P. Vicat-Blanc, "Sifran: Evaluating iot networks with a no-code framework based on ns-3," in 2022 Latin America Networking Conference (LANC),. ACM, 2022.

[58] P. Serrano, A. Garcia-Saavedra, G. Bianchi, A. Banchs, and A. Azcorra, "Per-frame energy consumption in 802.11 devices and its implication on modeling and design," IEEE/ACM Transactions on networking, vol. 23, no. 4, pp. 1243–1256, 2014.

[59] A. Sen, A. Mondal, B. Palit, J. Jayatheerthan, K. Paul, and S. Chakraborty, "An ns3-based energy module of 5g nr user equipments for millimeter wave networks," in IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2021, pp. 1–2.