

THE INTERNET OF METAMATERIAL THINGS AND THEIR SOFTWARE ENABLERS

Christos Liaskos^{1,2}, Georgios G. Pyrialakos^{2,3}, Alexandros Pitolakis^{2,3}, Ageliki Tsioliaridou², Michail Christodoulou^{2,3}, Nikolaos Kantartzis^{2,3}, Sotiris Ioannidis^{4,2}, Andreas Pitsillides⁵, Ian F. Akyildiz⁵

¹Computer Science Engineering Dept., University of Ioannina, Ioannina, Greece, ²Foundation for Research and Technology - Hellas, Heraklion, Greece, ³Electrical and Computer Engineering Dept., Aristotle University, Greece, ⁴Technical University of Chania, Crete, Greece, ⁵Computer Science Dept., University of Cyprus, Nicosia, Cyprus,

NOTE: Corresponding author: Christos Liaskos, cliaskos@uoi.gr

Abstract – A new paradigm called the Internet of MetaMaterial Things (IoMMT) is introduced in this paper where artificial materials with real-time tunable physical properties can be interconnected to form a network to realize communication through software-controlled electromagnetic, acoustic, and mechanical energy waves. The IoMMT will significantly enrich the Internet of Things ecosystem by connecting anything at any place by optimizing the physical energy propagation between the metamaterial devices during their lifetime, via “eco-firmware” updates. First, the means for abstracting the complex physics behind these materials are explored, showing their integration into the IoT world. Subsequently, two novel software categories for the material things are proposed, namely the metamaterial Application Programming Interface and Metamaterial Middleware, which will be in charge of the application and physical domains, respectively. Regarding the API, the paper provides the data model and workflows for obtaining and setting the physical properties of a material via callbacks. The Metamaterial Middleware is tasked with matching these callbacks to the corresponding material-altering actuations through embedded elements. Furthermore, a full stack implementation of the software for the electromagnetic metamaterial case is presented and evaluated, incorporating all the aforementioned aspects. Finally, interesting extensions and envisioned use cases of the IoMMT concept are discussed.

Keywords – Internet of Things, metamaterials, programming interface, software-defined networking.

1. INTRODUCTION

Recent years have witnessed the advent of the Internet-of-Things (IoT), denoting the interconnection of every electronic device and the smart, orchestrated automation it entails [1]. Vehicles, smart phones, sensors, home and industrial appliances of any kind expose a functionality interface expressed in software, allowing for developers to create end-to-end workflows. As an upshot, smart buildings and even smart cities that automatically adapt, e.g., power generation, traffic and heat management to the needs of residents, have been devised in recent years. This current IoT potential stems from exposing and controlling a high-level functionality of an electronic device, such as turning on/off lights and air-conditioning units based on the time of day and temperature. This paper proposes the expansion of the IoT to the level of physical material properties, such as electrical and thermal conductivity, mechanical elasticity, and acoustic absorption. This novel direction is denoted as the *Internet of MetaMaterial Things (IoMMT)*, and can have groundbreaking potential across many industrial sectors, as outlined in this paper. There are two key enablers for the proposed IoMMT:

Key Enabler 1:

The first key enabler of the proposed IoMMT are the metamaterials, the outcome of recent research in physics

that has enabled the creation of artificial materials with real-time tunable physical properties [2, 3]. Metamaterials are based on the fundamental idea stating that the physical properties of matter stem from its atomic structure. Therefore, one can create artificially structured materials (comprising sufficiently small elementary “units” of composition and geometry) to yield any required energy manipulating behavior, including types not found in natural materials. Metamaterials manipulating electromagnetic (EM) energy were the first kind of metamaterials to be studied in depth, mainly due to the relative ease of manufacturing as low-complexity electronic boards [3].

Going beyond EM waves, the collectively termed elastodynamic metamaterials can manipulate acoustic, mechanical and structural waves, whereas thermodynamic and quantum-mechanic metamaterials have also been postulated [4]. Elastodynamic metamaterials, empowered by recent advances in nano- and micro-fabrication (e.g. additive manufacturing/3D printing), can exhibit effective/macroscopic nonphysical properties such as tunable stiffness and absorption/reflection, extreme mass-volume ratios, negative sonic refraction, etc [5]. Their cell-size spans several length scales, depending on the application: acoustic cloaking/anisotropy/isolation, ultra-lightweight and resilient materials, devices for medical/surgical applications and food/drug adminis-

tration, MEMS, anti-seismic structures, etc. Tunability of elastodynamic metamaterials can be achieved with electric, magnetic, optical, thermal or chemical stimuli. In a nutshell, their operation is as follows: Impinging EM waves create inductive currents over the material, which can be modified by tuning the actuator elements within it (e.g., simple switches) accordingly. The Huygens principle states that any EM wavefront departing from a surface can be traced back to an equivalent current distribution over a surface [3]. Thus, in principle, metamaterials can produce any custom departing EM wave as a response to any impinging wave, just by tuning the state of embedded switches/actuators. Such EM interactions are shown in Fig. 1 (on the right side). The same principle of operation applies to mechanical, acoustic and thermal metamaterials [6].

Key Enabler 2:

The second key enabler of the IoMMT is the concept of networked metamaterials. These will come with an application programming interface (API), an accompanying software middleware and a network integration architecture that enable the hosting of any kind of energy manipulation over a metamaterial in real time (e.g., steering, absorbing, splitting of EM, mechanical, thermal or acoustic waves), via simple software callbacks executed from a standard PC (desktop or laptop), while abstracting the underlying physics. The goal is to constitute the IoMMT directly accessible to the IoT and software development industries, without caring for the intrinsic and potentially complicated physical principles. Regarding the IoMMT potential, large scale deployments of EM metamaterials in indoor setups have introduced the groundbreaking concept of programmable/intelligent wireless environment (Fig. 2) [7]. By coating all major surfaces in a space (e.g., indoors) with EM metamaterials, the wireless propagation can be controlled and customized via software. As detailed in [7] this can enable the mitigation of path loss, fading and Doppler phenomena, while also allowing waves to follow improbable air-routes to avoid eavesdroppers (a type of physical-layer security). In cases where the device beamforming and the EM metamaterials in the space are orchestrated together, intelligent wireless environments can attain previously unattainable communication quality and wireless power transfer [7]. Extending the EM case, we envision the generalized IoMMT deployed as structural parts of products, as shown in Fig. 3:

- EM interference and unwanted emissions can be harvested by IoMM-coated walls and be transformed back to usable EM or mechanical energy.
- Thermoelectric and mechanical metamaterials can micro-manage emanated heat and vibrations from devices, such as any kind of motor, to recycle it as energy while effectively cooling it. The same principle can be applied to a smart household or a noisy factory.

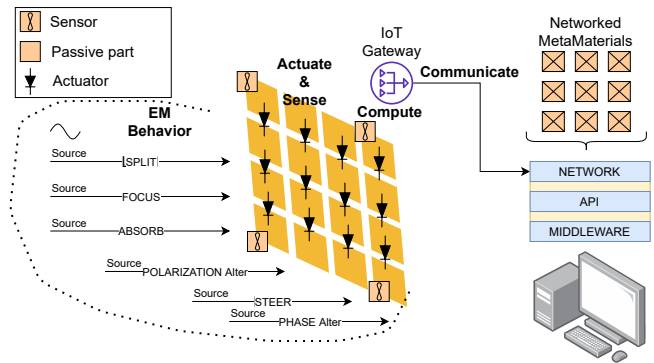


Fig. 1 – Networked metamaterial structure and possible energy wave interactions [8].

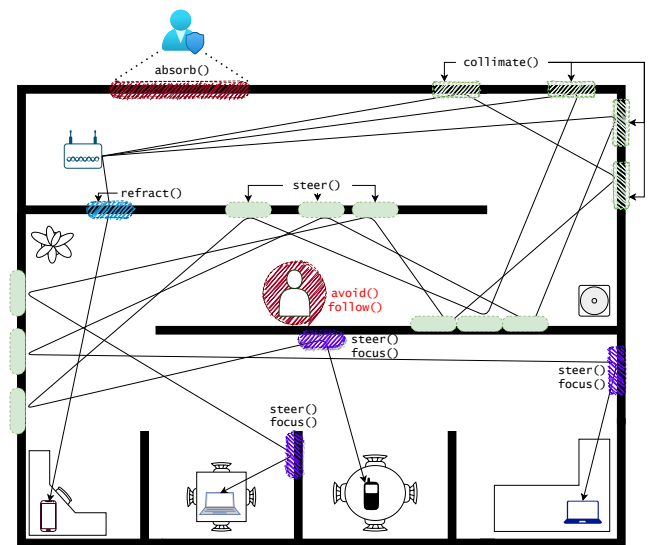


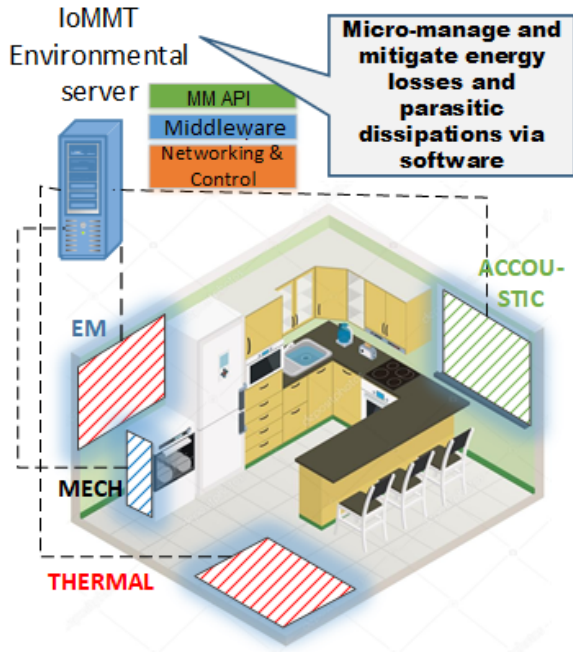
Fig. 2 – The programmable wireless environment introduced in [7], is created by coating walls with networked metamaterials. This allows for customized wireless propagation-as-an-app per communicating device pair, introducing novel potential in data rates, communication quality, security and wireless power transfer.

- The acoustic metamaterials can surround noisy devices or be applied on windows to provide a more silent environment, but to also harvest energy which can be added to a system such as a smart-household.

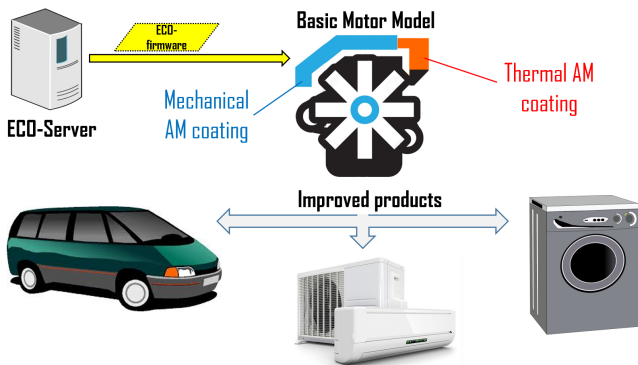
Assuming a central controller to optimize a given IoMMT deployment allows for further potential. For instance, one can allow for quickly “patching” of overlooked physical aspects (e.g., poor ecological performance) of IoMM-enabled products during operation, without overburdening the product design phase with such concerns. The “patching” may also be deferred in the form of “eco-firmware”, distributed via the Internet to ecologically tune a single product or horizontal sets of products.

In this context, the principal contributions of the paper are as follows:

- We propose the concept of the IoMMT and discuss



(a) Conceptual IoMMT deployment within a smart home.



(b) Conceptual IoMMT deployment within single products.

Fig. 3 – Envisioned applications of the IoMMT in smart houses and products.

its architecture and interoperability with existing network infrastructures.

- We define two novel categories of software: the *Metamaterial API* and the *Metamaterial Middleware*, which enable any software developer to interact with a set of networked metamaterials, in a physics-agnostic manner. We establish the data models, workflows, and test bed processes required for profiling and, subsequently, componentizing metamaterials.
- We present an implemented and experimentally verified version of the metamaterial API and the Metamaterial Middleware for the EM case.
- We highlight promising, new applications empowered by the featured IoMMT concept.

In this aspect, the potential of our IoMMT paradigm is the first to offer true control over the energy prop-

agation within a space, in every physical domain, i.e., for any physical material property and corresponding information-carrying wave. For instance, control over the equivalent RLC parameters of an electric load controls the power that can be delivered to it by an EM wave. Moreover, the presented software is a mature prototype platform for the development of IoMMT applications. This constitutes a major leap towards a new research direction. On the other hand, other research directions have proposed and explored the Internet of NanoThings [9]. Although similarly named, these directions are not related to the IoMMT, as they are about embedding nano-sized computers into materials in order to augment the penetration level of applications (e.g., sense structural, temperature, humidity changes within a material, rather than just over it, etc.), and not to control the energy propagation within them.

The remainder of this paper is organized as follows, devoting a section to each of the principal contributions of our work. In Section 2 we provide the related work overview and the necessary prerequisite knowledge for networked metamaterial. In Section 3 we present the architecture for integrating the IoMMT in existing Software-Defined Networks (SDNs) and systems. In Section 4 we present the novel metamaterial API, and Section 5 follows with the description of the Metamaterial Middleware and its assorted workflows. In Section 6 we present the implemented version of the software for the EM metamaterial case, along with a description of the employed evaluation test bed. Finally, novel realistic applications enabled with our new paradigm are discussed in Section 7, and we conclude the paper in Section 8.

2. PREREQUISITES AND RELATED WORK

Metamaterials are simple structures that are created by periodically repeating a basic structure, called a *cell* or a *meta-atom* [3]. Some examples across physical domains are shown in Fig. 4. The planar (2D) assemblies of meta-atoms, known as *metasurfaces*, are of particular interest currently [10, 11]. For instance, EM are currently heavily investigated by the electromagnetic/high-frequency community, for novel communications, sensing and energy applications. [12–14].

A notable trait of metamaterials is that they are simple structures and, therefore, there exists a variety of techniques for generally low-cost and scalable production [3]. The techniques such as printed circuit boards, flexible materials such as Kapton, 3D printing, Large Area Electronics, bio-skins and microfluidics have been successfully employed for manufacturing [3].

In each physical domain, a properly configured metamaterial has the capacity to steer and focus an incoming energy wave towards an arbitrary direction or even completely absorb the impinging power. In the EM case this capability can be exploited for advanced wireless communications [7, 15–19], offering substantially increased

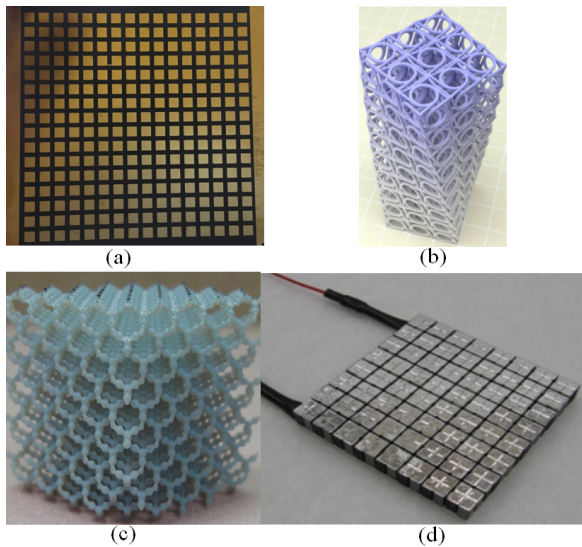


Fig. 4 – Energy manipulation domains of artificial materials: (a) Electromagnetic [20] (b) Mechanical [21] (c) Acoustic [22] (d) Thermoelectric [23].

bandwidth and security between two communicating parties.

The potential stemming from interconnected metamaterials has begun to be studied only recently [8]. The perspective networking architecture and protocols [7,8], metamaterial control latency models [24], and smart environment orchestration issues have been recently studied for the EM case [25,26].

Notably, a similarly named concept, i.e., the Internet of NanoThings [9], was recently proposed to refer to materials with embedded, nano-sized computing and communicating elements. In general, these materials are derived from miniaturizing electronic elements and placing them over or embedding them into fabrics and gadgets, to increase their application-layer capabilities. For instance, this could make a glass window become a giant, self-powered touchpad for another IoT device. Originally, the concept of software-defined metasurfaces was based on the nano-IoT as the actuation/control enabler [17]. Nano-devices can indeed act as the controllers governing the state of the active cells, offering manufacturing versatility and extreme energy efficiency. Nonetheless, until nano-IoT becomes a mainstream technology, other approaches can be adopted for manufacturing software-defined metasurfaces, as reported in the related physics-oriented literature [6]. It is also noted that nano-IoT as a general concept is about embedding nano-sized computers into materials in order to augment the penetration level of applications (e.g., sense structural, temperature, humidity changes within a material, rather than just over it, etc.), and not specifically to control the energy propagation within them.

In contrast, our work refers specifically to the case of metamaterials and the capabilities they offer for the manipulation of energy across physical domains. Moreover, our paper introduces the software enablers for this direction, which has not been proposed before. Additionally,

our paper focuses more on the networking approaches for metamaterials, which has only been treated in our previous work [8], and only for the EM case. Finally, the work of Chen et al. [27] also advocates for the use of metamaterial in any physical domain for distributed energy harvesting, e.g., in a smart house or a city. However, software enablers and networking considerations are not discussed or solved in [27]. Moreover, the energy manipulation type is restricted to harvesting which can be viewed as a subset of our proposed IoMMT potential.

Metamaterials: Principles of Operation, Classification and Supported Functionalities

A conceptual metamaterial is illustrated in Fig. 5 [3]. Basically, a metamaterial consists of periodically repeated meta-atoms arranged in a 3D grid layout, with the metasurfaces being a sub-case. In particular, unit cells comprise passive and tunable parts, required in reconfigurable metamaterials as well as optional integrated sensory circuits, which can extract information of the incident energy wave. Furthermore, tunable parts are crucial for metamaterials, as they enable reconfigurability and switching between different functions. For illustration, in EM metamaterials at microwave frequencies, the tunable parts embedded inside the unit cells can be voltage-controlled resistors (varistors) and/or capacitors (varactors), micro-electromechanical switches (MEMS), to name a few [3,19].

On the other hand, in mechanical and acoustic metamaterials, the tunable parts can be micro-springs with a tunable elasticity rate [28,29]. The meta-atoms may also form larger groups, called *super-atoms* or *super-cells*, repeated in specific patterns that can serve more complex functionalities, as discussed later in this paper. Lastly, the software-defined metamaterials include a *gateway* [7], i.e., an on-board computer, whose main tasks are to: i) power the whole device and ii) control (get/set) the state of the embedded tunable elements, iii) interoperate with the embedded sensors, and iv) interconnect with the outside world, using well-known legacy networks and protocol stacks (e.g. Ethernet).

The relative size of a meta-atom compared to the wavelength of the excitation (impinging wave) defines the energy manipulation precision and efficiency of a metamaterial. For example, EM metasurfaces share many common attributes with classic antenna-arrays and reflect-arrays. Antenna arrays can be viewed as independently operating antennas, being very effective for coarse beam steering as a whole. Reflect-arrays typically consist of smaller elements (still subwavelength), permitting more fine-grained beam steering and a very coarse polarization control. Metamaterials comprise orders of magnitude smaller meta-atoms, and may also include tunable elements and sensors. Their meta-atoms are generally considered tiny with regard to the exciting wavelength, hence allowing full control over the form of the departing energy wave.

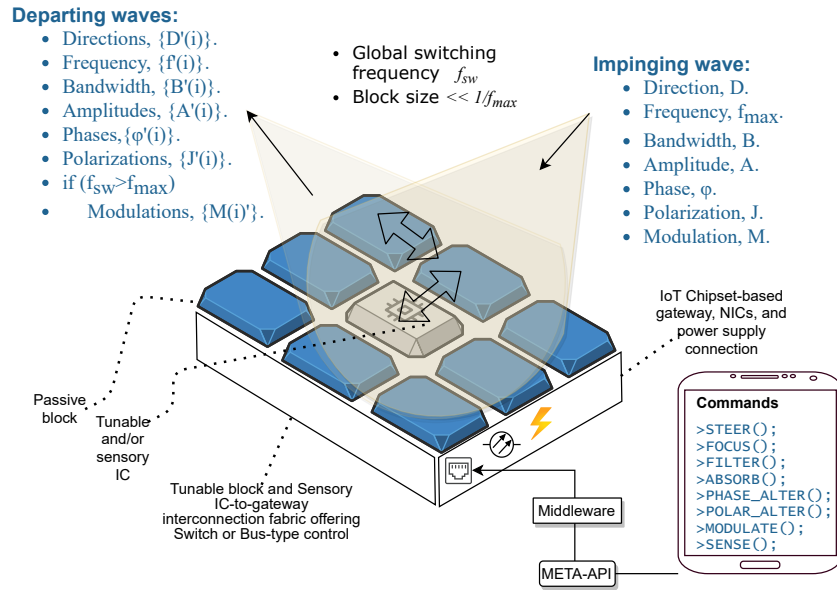


Fig. 5 – Overview of the metasurface/metamaterial structure and operating principles.

Regardless of their geometry and composition, the operating principle of metamaterials remains the same. As depicted in Fig. 5, an impinging wave of any physical nature (e.g., EM, mechanical, acoustic, thermal) excites the surface elements of a metamaterial, initiating a spatial distribution of energy over and within it. We will call this distribution “exciting-source”. On the other hand, well-known and cross-domain principles state that any energy wavefront, which we demand to be emitted by the metamaterial as a response to the excitation, can be traced back to a corresponding surface energy distribution denoted as “producing-source” [3, 6]. Therefore, a metamaterial configures its tunable elements to create a circuit that morphs the exciting-source into the producing-source. In this way, a metamaterial with high meta-atom density can perform any kind of energy wave manipulation that respects the energy preservation principle. Arguably, the electromagnetism constitutes a very complex energy type to describe and, as a consequence, manipulate in this manner, as it is described by two dependent vectors (electric and magnetic field) as well as their relative orientation in space, i.e. polarization (mechanical, acoustic and thermal waves can be described by a single scalar field in space). As such, incoming EM waves can be treated in more ways than other energy types. The common types of EM wave manipulation via metamaterials, reported in the literature [3], can designate a set of high-level functionality types as follows:

- Amplitude: Filtering (band-stop, -pass), absorption.
- Polarization: Waveplates (polarization conversion, modulation).
- Wavefront: Steering (reflecting or refracting), split-

ting, focusing, collimating, beamforming, scattering.

- Bandwidth: Filtering.
- Modulation: Requires embedded actuators that can switch states fast enough to yield the targeted modulation type [30].
- Frequency: Filtering, channel conversion.
- Doppler effect mitigation and non-linear effects [8].

Additionally, sensing impinging waves may be considered one of the above functionalities and, as an outcome, the embedded sensors can extract information of any of the above parameters related to the incident wave.

In this aspect, the role of the contributed metamaterial API is to model these manipulation types into a library of software callbacks with appropriate parameters. Then, for each callback and assorted parameters, the Metamaterial Middleware produces the corresponding states of the embedded tunable elements that indeed yield the required energy manipulation type. In other words, a metamaterial coupled with an API and a Metamaterial Middleware *can be viewed as a hypervisor that can host metamaterial functionalities* upon user request [8].

In the following, we focus on EM metamaterials which, as described, yield the richest API and most complex Metamaterial Middleware. The expansion to other energy domains is discussed via derivation in Section 7.

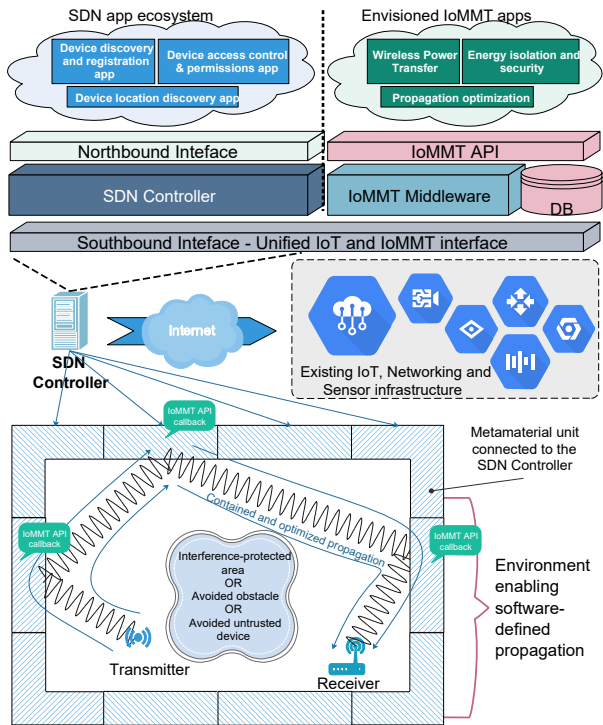


Fig. 6 – SDN schematic display of the system model and the entire workflow abstraction.

3. NETWORKED METAMATERIALS AND SDN WORKFLOWS

Many metamaterials deployed within an environment can be networked through their gateways. This means that they may become centrally monitored and configured via a server/access point in order to serve a particular end objective.

An example is given in Fig. 6, where a set of metamaterials is designed with the proper commands for energy wave steering and focusing, in order to route the energy waves exchanged between two wireless users, thus avoiding obstacles or eavesdroppers. Other applications include wireless power transfer and wireless channel customization for an advanced quality of service (QoS) [7, 13]. Such a space, where energy propagation becomes software defined via metamaterials is called a programmable wireless environment (PWE) [8].

As shown in [7], the PWE architecture is based on the software-defined networking (SDN) principles. The PWE server is implemented within an SDN controller [31]; the southbound interface abstracts the metamaterial hardware, treating metamaterial devices as networking equipment that can route energy waves (e.g. similar to a router, albeit with a more extended and unique parameterization). Thus, the metamaterial API constitutes a part of the northbound SDN interface, atop of which the security, QoS and power transfer concepts can be implemented as SDN controller applications. On the other hand, the Metamaterial Middleware is part of the SDN middleware, translating metamaterial

API callbacks into metamaterial hardware directives. A notable trait of the Metamaterial Middleware is that it is divided into *two parts*, in terms of system deployment [32]:

1. The metamaterial manufacturing stage component, a complex, offline process requiring special metamaterial measurement and evaluation setups (discussed in Section 5), and
2. The metamaterial operation stage component, which operates in real time based on a codebook. This codebook is a database populated once by the manufacturing stage component and contains a comprehensive set of configurations for all metamaterial API callbacks, supported by a given metamaterial.

The operation stage component simply retrieves configurations from the codebook and optionally combines them as needed, using an interleaving process described in Section 5.

Notably, other studies propose the use of online machine learning as a one-shot process, which can be more practical when response time is not a major concern [33]. However, in this work we propose the aforementioned separation in deployment, to ensure the fastest operation possible overall, thus covering even the most demanding cases.

It is noted that SDN is not a choice due to restrictions, but rather a choice due to compatibility. In the software-defined metasurfaces presented in this paper, a key point is the abstraction of physics via an API that allows networking logic to be reused in PWEs, without requiring a deep understanding of physics. SDN has (among other things) already introduced this separation of control logic from the underlying hardware and its administrative peculiarities. Therefore, we propose an integration of PWE within SDN to better convey the logical alignment of the two concepts.

4. APPLICATION PROGRAMMING INTERFACE FOR METAMATERIALS

In the following, we consider a metamaterial in the form of a rectangular *tile*. The term *tile* is used to refer to a practical metamaterial product unit, which can be used to cover large objects such as walls and ceilings in a floorplan.

A software process can be initiated for any metamaterial tile supporting a unique, one-to-one correspondence between its available switch element configurations and a large number of metamaterial functionalities. The metamaterial tiles in this work incorporate tunable switch elements, which dictate the response of each individual cell, locally. In this way, providing an arrangement of all the tile cells allows the tuning of the “concerted” metamaterial response of the entire tile.

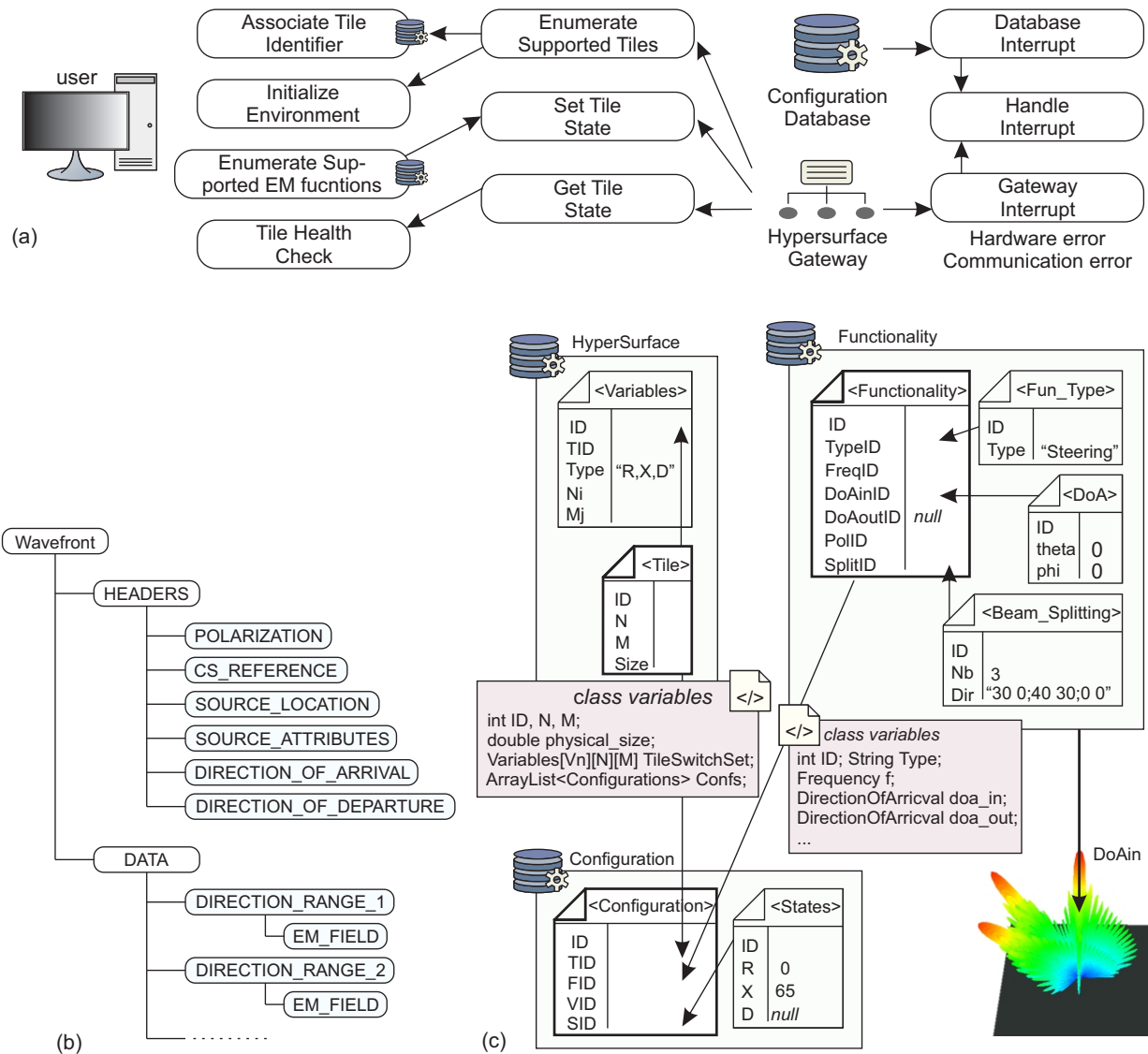


Fig. 7 – (a) Case diagram of the main functions supported by the three basic entities. Tasks highlighted with the database icon indicate that a set of data is to be retrieved from the Configuration Database. (b) Wavefront description in data object format. (c) A simplified overview of the structure of the Configuration Database. The <Functionality> table combines a set of metamaterial parameters to define new functionalities. Both tables are combined in the <Configuration> table with a set of entries from the <States> table to compose a new configuration that supports the functionality FID on tile TID (VID refers to an entry in the table <Variables>).

In this section, we present the API that grants access to the tile's metamaterial applications by defining an abstract representation of the metamaterial, its switch element configurations and their respective functionalities. Specifically, the API resides between a user, operating a common PC (desktop, smartphone, etc.) and a tile gateway, linked to the network of switch element controllers. The case diagram of the proposed concept, presented in Fig. 7(a), involves the following main entities:

- A Configuration Database which stores all information regarding the tiles, the switch element configurations, and their corresponding functionalities.
- A User which initiates all API callbacks through either the source code or button click events in the Graphical User Interface (GUI).
- The HyperSurface Gateway which represents the electronic controller of the hardware.
- An Interrupt handling service which acts as a persistent daemon, receiving and dispatching commands to the Hypersurface Gateway.

4.1 Data Structures of the Metamaterial API

In the configuration Database (DB), each tile is associated with an element array S that represents all possible arrangements of switch element states on the metamaterial under study. Each switch element is represented by either a discrete or a continuous variable, creating a mathematical space of $V_1 \times N \times M + \dots + V_n \times N \times M$ dimensions, where V_i is the number of elements of the same type (e.g. capacitors) and N, M the number of unit cells towards the two perpendicular directions. Furthermore, every object in this space corresponds to a different state of S and therefore a different configuration. As an example, a tile with two controllable resistive and one diode elements per unit cell is parameterized by a $2 \times N \times M + 1 \times N \times M$ array, where the first and second sets span a continuous $[R_{\min}, \dots, R_{\max}]$ and a discrete $[0, 1]$ range, respectively. In this case, 0 and 1 correspond to the OFF and ON states of the diode. This representation will then acquire the following form

$$F \leftarrow [(d_1, d_2, i_1)_{n=1}, \dots, (d_1, d_2, i_1)_{n=N \times M}] \quad (1)$$

where d_1, d_2 are double-type variables, i_1 is an integer variable, and F is the appointed functionality. The primitive data types of all variables should be selected so as to minimize the total parameter space of combined states without any loss of relevant information. This lays a better optimized communication and computational burden to both the API and the Compiler, especially during the compilation process where a sizable amount of mathematical computations is required. Accordingly, all functionalities are, also, associated with their own representation and classified pertinent to their own type and defining parameters. For instance, a complex beam-splitting and polarization control operation

is parameterized by a discrete variable corresponding to the number of outgoing beams, their directivity amplitudes, and an appropriate number of (θ, φ) pairs, indicating the steering angles. The most complex functionality can be generally described by a custom scattering pattern and represented herein by a collection of variables that indicate the reflected power towards all directions within the tile's viewing area.

It is, also, worth noting that the data objects being passed as arguments in the callbacks are primarily descriptions of wavefronts. A simplified data structure is illustrated in Fig. 7(b). Hence, a wavefront is described by a type (string identifier), such as "Planar", "Elliptic", "Gaussian", "Custom", etc. For each type, a series of headers defines the location and attributes of the creating source (for impinging wavefronts only) as well as the coordinate system origin with respect to which all distances are measured. Moreover, the direction of arrival and departure are arrays that can be used to define multiple impinging or departing wavefronts at the same time. Notably, the information within the headers may be sufficient to produce any value of the wavefront via simply analytical means. In such cases, the data part can be left empty. In custom wavefronts, the data is populated accordingly. A mechanism for defining periodicity is supplied via the notion of ranges (i.e. coordinate ranges where the energy field is approximately equal), to potentially limit the size of the overall data object.

The parameters that represent the functionalities and configurations of a tile constitute the set of variables that are exposed to the programmer through the metamaterial API. They are organized in a unified manner within the Database, as shown in Fig. 7(c) which provides an illustration of the unique association between all primary tables. Particularly, the <Tile> table stores all information of a tile's hardware implementation, such as the number of variables per unit cell and the type of switch elements. The <Functionality> table stores the representation scheme described in subsection 2 for all available metamaterial functionalities. Each parameter associated with a functionality is organized in a separate table, including a table that stores an identification variable representing the type of functionality. This table ID enumerates all possible operations supported by the tile, including full power absorption, wavefront manipulation (steering, splitting, etc.), and wavefront sensing. Finally, the <Configuration> table combines, in an exclusive manner, both primary tables (<Tile> and <Functionality>) to link each stored functionality with a specific set of switch element states, acquired from the pool of available entries in the secondary table <States>.

4.2 API Callbacks and Event Handling

Using the Database as a reference point, the API is responsible for interpreting a configuration array to the

proper set of hardware commands, when a suitable callback is executed. In general, an API callback can refer to a number of common requests such as:

- Detect the number and type of accessible tiles in the environment.
- Get the current state of all switch elements or set them to a specific configuration.
- Check the health status and handle interrupts from the tiles or the Database.

Prior to any other callback, the API follows an initiation process, while the software detects all presently active and connected (discoverable) tiles by broadcasting a corresponding network message. The tiles report their location and a unique identifier, e.g. a fixed value, that associates all tiles with the same hardware specifications. The API validates the support of the active tiles by checking if the identifier exists in the tile list present in the database. It, then, retrieves the switch element arrays that correspond to these tiles and remains idle until a new “get” or “set” request is received for a currently active or new configuration, respectively. This means that the API is now open to receive new functionality requests from a user, physically operating the software, or generate its own requests by reacting to unexpected changes in the environment of devices linked to the MS network. When a new functionality request is received, the API retrieves one of the available configurations from the Database and translates it to a proper set of element states on an active metamaterial tile. The corresponding API callback process is illustrated in Fig. 8. In particular, the Caller (user) executes a metamaterial Function Deployment request, which, in turn, invokes the Configuration Resolver, identifying a tile that supports the requested functionality. Next, the resolver queries the Database and returns a configuration that matches the intended metamaterial application, looking for a proper entry in the <Configuration> table. The API creates a string command, using the tile identifier and a hardware representation

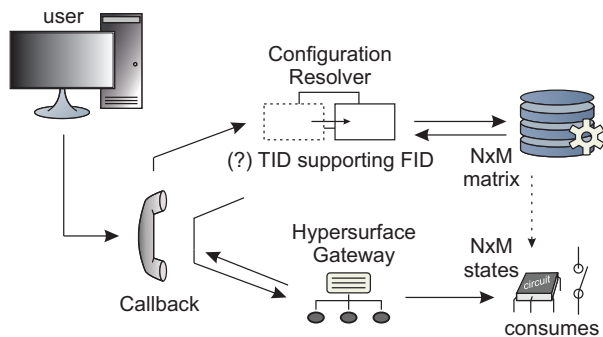


Fig. 8 – A metamaterial Function Deployment request initiates an API callback on the tile (TID). The Configuration Resolver seeks an appropriate configuration that supports the selected functionality (FID). The matrix of states is conveyed to the HyperSurface Gateway, where it is translated to a set of corresponding hardware states.

of the element state variables, which is, then, conveyed to the tile Gateway using the corresponding protocol. This notifies the intra-tile control network to assign the switch element states to their suitable values. Finally, feedback from a successful or failed configuration setup is received from the tile, notifying the user. The state of the newly set configuration is evaluated through either the identification of failed or unresponsive switches, or by activating the sensing app, in controlled conditions, as a self-diagnosing tool for the tile.

A more advanced API callback may involve the assignment of a secondary or supplementary functionality, on top of an already existing operation. For instance, the Metamaterial Middleware may receive independent requests from different users to steer the wavefront of several point power sources (i.e. the users’ cellphones) towards the direction of a single nearby network hotspot. This can be handled by the API in many ways. In the case where several tiles are present in the environment, each tile can be repurposed to host a separate functionality, distributing all users to their own active tiles. When this is not feasible, the API can divide a single tile into separate areas and associate the respective element switches to different configurations (the division is usually performed in equal-sized rectangular patterns, but interlacing can, also, be used). Lastly, two functionalities can be combined into a single one, when a corresponding physical interpretation exists. For example, two separate steering operations, from the same source, may be combined into a single dual-splitting operation, expressed by a single unified pattern on the metamaterial.

In other cases, the configuration resolver may need to combine several functionalities to produce a special new application. This occurs when a functionality is parameterized by a continuous variable (e.g. a steering angle), while the Metamaterial Middleware can evaluate and store only a finite number of entries in the database. In such a scenario, the resolver seeks the two closest matching entries through an appropriate minimization function (e.g. for a beam-steering operation, the minimum distance between the requested and currently stored steering angles), whereas performing an interpolation of the switch state values.

To ensure a seamless operation, the API is reinforced with a set of specialised algorithms to handle unexpected failures in the hardware or communication network. So, a tile must include all necessary identification capabilities (e.g. the ability to identify power loss in its switch elements), and be able to notify the Metamaterial Middleware in case of failure. The API is, then, responsible for handling these errors by ensuring no loss of the current functionality [32]. For illustration, if a group of switch elements is stuck to an unresponsive state, the Metamaterial Middleware may instantly seek the closest matching functionality with a fixed configuration for the faulty elements. In more severe cases of demanding human intervention, like an unresponsive network (af-

ter a certain timeout), the API is also responsible for informing an available user.

5. THE METAMATERIAL MIDDLEWARE

For the metamaterial to be reconfigured between different functionalities, a physical mechanism for locally tuning each unit cell response must be infused [8]. In the context of the present work, we assume that the response of the unit cells is controlled by variable *impedance loads* connected to the front side metallization layer of the metamaterial, where structures such as the resonant patch pair resides [3]. The loads are complex valued variables, comprising resistors and capacitors or inductors. The value of the i -th load, $Z_i = R_i + jX_i$, comprises two parameters: its resistance ($R_i > 0$) and reactance ($X_i = -(\omega C_i)^{-1}$ or $X_i = +\omega L_i$), for capacitive and inductive loads, respectively. The loads are, thus, electromagnetically connected to the surface impedance of the “unloaded” unit cell and by tuning their values we can regulate the unit cell response, e.g. the amplitude and phase of its reflection coefficient. The latter is naturally a function of frequency and incoming ray direction and polarization. When the metamaterial unit cells are properly “orchestrated” by means of tuning the attached (R_i, X_i) loads, the desired functionality (global response) of the metamaterial is attained.

In the most rigorous approach, the metamaterial response can be computed by full-wave simulations, which implement Maxwell’s laws, given the geometry and metamaterial properties of the structure as well as a complex vector excitation, i.e. the impinging wave polarization and wavefront shape (phase and amplitude profile). The full-wave simulation captures the entire physical problem and, hence, does not require a metamaterial-level abstraction for the structure. Frequency-domain solvers, which assume linear media and harmonic excitation (i.e. the same frequency component in both the excitation and the response), are the prime candidates for full-wave simulation. They typically discretize the structure’s volumes or surfaces at a minimum of $\lambda/10$ resolution, formulate the problem with an appropriate method (e.g. the finite-element or the boundary-element method) and, then, numerically solve a large sparse- or full-array system to compute the response, in our case, the scattered field. Conversely, time-domain solvers assume a pulsed excitation, covering a predefined spectral bandwidth and iteratively propagate it across the structure, solving Maxwell’s equations to compute its response; they, typically, require a dense discretization of the structure, e.g. a minimum of $\lambda/20$ resolution. From this process, it becomes evident that metamaterial with a wide aperture, i.e. spanning over several wavelengths along the maximum dimension, require high computational resources in the full-wave regime, scaling linearly, when parametric simulations need to be performed to optimize the structure

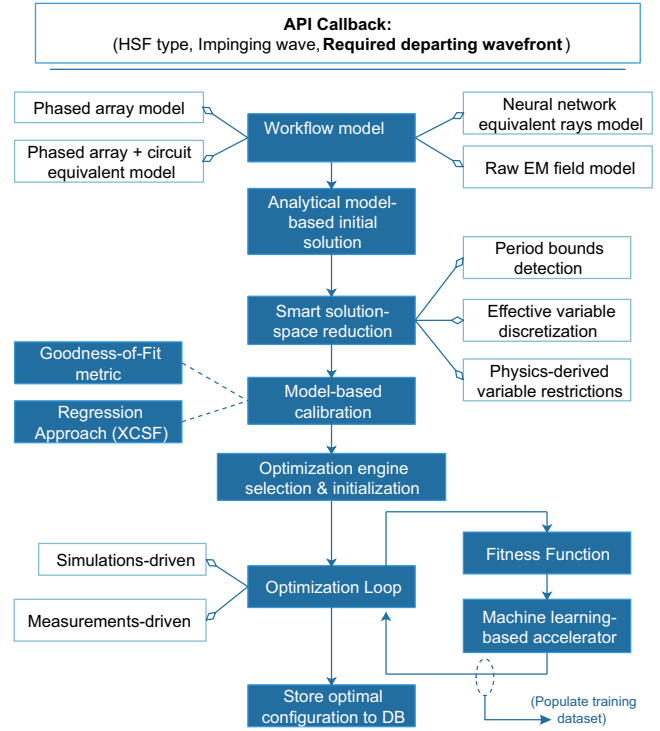


Fig. 9 – The Metamaterial Middleware functionality optimization workflow. The workflow seeks to match an analytical metamaterial model and its parameters to a specific parameterized API callback. A selected analytical model is first calibrated. Then, an iterative process (simulation or measurement-based) optimizes the input parameters of the model that best yield the API callback.

and/or the response. For instance, broadband simulation for the response of a unit cell of volume $(\frac{\lambda}{5})^3$ on a contemporary desktop computer could take several minutes, especially if the cell includes fine subwavelength features. The memory and CPU resources scale-up linearly for metasurfaces comprising hundreds of thousands of unit cells. Moreover, full-wave simulations do not explicitly unveil the underlying principles that govern the metamaterial functionality.

5.1 Functionality Optimization Workflow: Metamaterial Modeling and State Calibration

In this section we establish the optimization workflow of Fig. 9 that drives the calibration process of the metamaterial via an appropriate approximation model. Here calibration denotes the matching of actual active element states (e.g., the states of a tunable varactor) to the corresponding model parameter values (e.g., phase difference per cell in the reflectarray model). In this workflow, the optical scattering response is initially investigated and, then, the solution is hill-climbed via an optimization loop relying on either field measurements or full-wave simulations. The approximate models are as follows.

The simplest model is the phased antenna-array analysis, where each single unit cell is treated as an independent antenna, excited by a single impinging ray and

emitting a single ray in response, with a local phase and amplitude alteration. Assuming a metamaterial consisting of $M \times N$ unit cells, the scattered E -field complex amplitude pattern at a given frequency can be calculated by the envelope (coherent superposition) of all rays scattered from the metamaterial [2]

$$E(\theta, \varphi) = \sum_{m=1}^M \sum_{n=1}^N A_{mn} e^{j\alpha_{mn}} f_{mn}(\theta_{mn}, \varphi_{mn}) \cdot \Gamma_{mn} e^{j\gamma_{mn}} f_{mn}(\theta, \varphi) e^{j\Phi_{mn}(\theta, \varphi)}. \quad (2)$$

In (2), φ and θ are the azimuth and elevation angles in the scattering direction, $(\theta_{mn}, \varphi_{mn})$ denotes the direction of the wavefront ‘ray’ incident on the mn -th cell, A_{mn} and α_{mn} are the amplitude and phase of the incident wavefront on the mn -th cell, Γ_{mn} and γ_{mn} form the reflection coefficient (amplitude and phase) of the mn -th cell, while f_{mn} defines the scattering pattern of the mn -th cell, which, according to reciprocity, is identical for the incident and scattered direction, and, in this work, is assumed that $f_{mn}(\theta, \varphi) = \cos(\theta)$. Finally, $\Phi_{mn}(\theta, \varphi)$ is the phase shift in the mn -th cell stemming from its geometrical placement, as

$$\Phi_{mn}(\theta, \varphi) = k \sin \theta [d_x m \cos \varphi + d_y n \sin \varphi] + \phi_0(\theta, \varphi), \quad (3)$$

where $d_{x,y}$ are the rectangular unit-cell lateral dimensions, $k = 2\pi/\lambda$ is the wavenumber in the medium enclosing the metamaterial, and ϕ_0 is the reference phase denoting the spherical coordinate system center, typically in the middle of the metamaterial aperture. Given a uniform, single-frequency impinging wave, any departing wavefront is essentially a Fourier composition of the individual meta-atom responses. Thus, we can, also, calculate the meta-atom amplitudes Γ_{mn} and phases γ_{mn} that yield a desired departing wavefront, by applying an inverse Fourier transform, as elaborately discussed in [2]. The calculated Γ_{mn} and γ_{mn} values must be mapped to the R_i and X_i values that generate them, since the latter are the actual tunable metamaterial parameters. This process requires a set of simulations yet it can be automated: existing model calibration techniques, such as the Regression and Goodness of Fit can be employed [34].

The shortcoming of the antenna-array approach is that the coupling between adjacent unit cells (e.g., compare against Fig. 5) is not properly accounted for, which can result to model imprecision [2]. To this aim, the Metamaterial Middleware user is presented with an alternative model. It utilizes the phased array and equivalent circuit model, which assumes not only the transmitting-responding antenna per meta-atom, but, also, circuit elements that interconnect them and account for the cross-meta-atom metamaterial interactions. The disadvantage of this approach is that an expert needs to define this circuit model, that is generally unique per metamaterial design [3]. Once this model has been selected and

provided in the proper format, the optimization workflow of Fig. 9 continues, once again, with the calibration phase, which is identical as before. The key difference and merit is that the calibration is, now, extremely precise with regard to the full-wave simulations, while it takes much less time to complete, as detailed in the corresponding study of [2].

An intermediate solution, combining the precision of the circuit model and the automation of the antenna-array model, is an equivalent propagation model, mentioned here for the sake of completion. The main idea is to introduce a generic mechanism to capture the cross-interactions among meta-atoms (as opposed to the strict, physics-derived nature of the circuit model) and then proceed with automatic model calibration, avoiding the need for expert input. The equivalent ray model uses a neural network approach as the generic cross-talk descriptor [35]. A short summary is as follows. Each meta-atom is mapped to a neural network node, and the locally impinging wave amplitude and phase are its inputs. Then, we clone this layer (omitting the inputs) and form a number of intermediate, fully connected layers (usually 3-5), thereby emulating a recurrent network with a finite number of steps. We define links per node (shared among all node clones), which define an alteration of the local phase and amplitude, and its distribution to other neighboring meta-atoms/nodes. Next, we proceed to calibrate the model via feed-forward/back-propagation, thereby obtaining a match between R , X , Γ_{mn} , and γ_{mn} values. Nonetheless, despite its automated nature, a major drawback of this model is the need for considerable computational resources, without which the model loses its value, since it becomes restricted only to very simple metamaterial designs.

Since computational complexity is a concern regardless of the chosen model, the Metamaterial Middleware workflow allows the user to define solution reduction across three directions. First, meta-atoms may be grouped into periodically repeated super-cells. Thus, the optimization workflow needs only to optimize the configuration parameters of a super-cell, as opposed to optimizing the complete metamaterial. Second, the range of possible R and X values per meta-atom can be discretized into regular or irregular steps, reducing the solution space further¹. Finally, some R and X values or ranges can be discarded due to the physical nature of the optimization request. For instance, if we seek to optimize a wave steering approach with an emphasis on minimal losses over the metamaterial (maximum reflection amplitude), the Ohmic resistance R needs to receive its boundary value. On a related track, machine learning-based approaches can quickly estimate the performance deriving from one set of R and X values, thereby discarding non-promising ones and accelerating convergence [36].

¹Notably, contemporary optimization engines already incorporate equivalents to this direction, as they are able to detect strongly and loosely connected inputs-outputs [34].

Subsequently, the Metamaterial Middleware workflow moves to the optimization stage, where it attempts to hill-climb the initial solution detected via any of the described approximate models. At this point, the workflow is compatible to any modern optimization engine, which receives an input solution and outputs one or more proposed improvements upon it at each iteration. Herein, we stress the existence of engines that, also, incorporate machine learning mechanisms, to accelerate the optimization cycle [34]. The optimization can be based either on full-wave simulations or a real measurement test bed, described in Section 6. The optimization metric can be any reduction of the produced departing wavefront. Various metrics relevant to antenna and propagation theory may be extracted, namely: the number of main lobes (beam directions), the directivity of main lobes, the side (parasitic) lobes and their levels, the beam widths, etc. Such metrics can be used to quantify the metamaterial performance for the requested functionality, e.g. the main lobe directivity and beam width measures how “well” a metamaterial steers an incoming wavefront to a desired outgoing direction. Lastly, the hill-climbed metamaterial configuration pertaining to the metamaterial API callback is stored into a database for any future use by metamaterial users.

Finally, we note that multiple simultaneous functionalities can be supported by interlacing different scattering profiles across the metamaterial. In general, this is performed by spatially mixing the profiles in phasor form

$$A_{mn}e^{j\alpha_{mn}} = \sum_{c=1}^{N_c} A_{c,mn}e^{j\alpha_{c,mn}}, \quad (4)$$

where c iterates over single, “low-level” functionalities and n, m are the unit cell indices. Typically, low-level functionalities correspond to simple beam steering operations, which are produced exclusively by phase variations on the metamaterial ($A_{c,mn} = 1$). In this case, a “high-level” functionality will correspond to a multi-splitting operation with variable spatial distribution of A_{mn} amplitude, raising the hardware requirements for the metamaterial. Therefore, a metamaterial with no absorption capabilities (and thus no control over A_{mn}) will have limited access to high-level operations, unless a mathematical approximation is to be applied, skewing the scattering response from its ideal state. As discussed in Section 6, a method for minimizing amplitude variations has been successfully investigated by increasing the number of secondary parasitic lobes. Such a problem can be easily reformulated into an optimization task, where an optimal match to the ideal high-level operation can be pursued under specific constrains (e.g. $A_{mn} > \text{const.}, \forall m, n$).

5.2 The Metamaterial Functionality Profiler

The optimization workflow of Fig. 9 opts for the best metamaterial configuration for a given, specific pair of

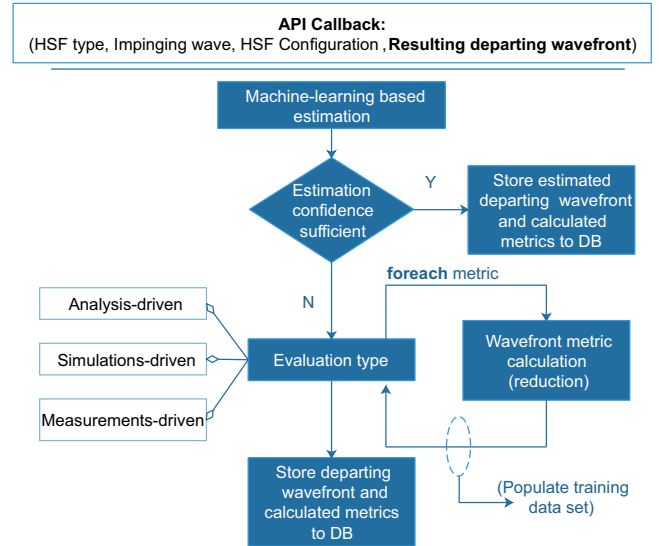


Fig. 10 – Workflow for profiling a metamaterial functionality. The workflow seeks to produce a data set that describes the metamaterial behavior for any impinging wave type that does not match the one specified in the current metamaterial configuration. An exhaustive evaluation takes place first for a wide set of possible impinging waves. For intermediate impinging wave cases, the workflow can rely on estimations produced by machine learning algorithms or simple extrapolation means, provided that it yields an acceptable degree of confidence.

impinging and departing wavefronts. However, in real deployments, it is not certain that a metamaterial will always be illuminated by the intended wavefront [24]. For instance, user mobility can alter the impinging wavefront in a manner that has limited relation to the intended one and, consequently, to the running metamaterial configuration. As such, there is a need for *fully profiling a metamaterial*, i.e. calculate and cache its expected response for each intended metamaterial configuration, but, also, for each possible (matching or not) impinging wavefront of interest. This profiling process is outlined in Fig. 10.

The profiling process begins by querying the existing cache (part of the DB) or trained model for the given metamaterial and an estimation (or existing calculated outcome) of the expected metamaterial response for a given impinging wave. If it exists, this response is stored into a separate profile entry for the metamaterial in the Metamaterial Middleware DB². If the response needs to be calculated anew, the process proceeds with either an analysis-, simulation- or measurement-driven evaluation. Therefore, the choice is given as a means to facilitate the expert into reducing the required computational time, as allowed per case. Then, the profiler proceeds to, also, calculate all possible reductions of the departing wavefront, e.g. the number of main lobes (beam directions), the directivity of the main lobes, the side (parasitic) lobes and their levels, the beam widths,

²In case of an estimated response, the user has control over the process to filter out estimations with low confidence. However, the selected estimation engine must be able to provide a confidence degree for this automation.

etc. Finally, once all required impinging wavefronts have been successfully processed, the profiling process is concluded.

It is clarified, that the middleware operations are one-time only, i.e., once the database containing the behavior profile of a metasurface is complete, it can be used in any application setting in the real world by any tile of the same type.

6. SOFTWARE IMPLEMENTATION AND EVALUATION

Employing the concepts of the previous sections, we developed a complete Java implementation of the described software. The software is subdivided into two integral modules: i) an implementation of the metamaterial API that handles the communication and allocation of existing configurations and ii) the Metamaterial Middleware that populates the configuration DB with new data (new tiles, configurations, and functionalities). The Metamaterial Middleware incorporates a full GUI environment, guiding the user through a step-by-step process to produce new configurations. It utilizes all available theoretical and computational tools for the accurate characterization of a metamaterial tile. Furthermore, it offers direct access to the configuration DB, manually, via a custom-made Structured Query Language (SQL) manager or through the automated process following a successful metamaterial characterization. Through this process, all the necessary data related to a newly produced configuration become explicitly available to the API.

A microwave metasurface was selected to demonstrate the capabilities of the developed concepts and methods for software-tunable metamaterials. We adapted the design of [37], where a set of RF diodes can be employed to toggle the reflection-phase of each cell between 16 states. We numerically extracted the response of the metasurface (i.e., its scattering pattern), and finally used the developed software to demonstrate how the metasurface response can be controlled. For the practical demonstration of the developed software in the same measurement environment (anechoic chamber) with a simpler, 1-bit metasurface hardware, we redirect the reader to [20,30], since the hardware manufacturing topic is quite extensive and clearly beyond the software aspects that constitute the focus of this paper.

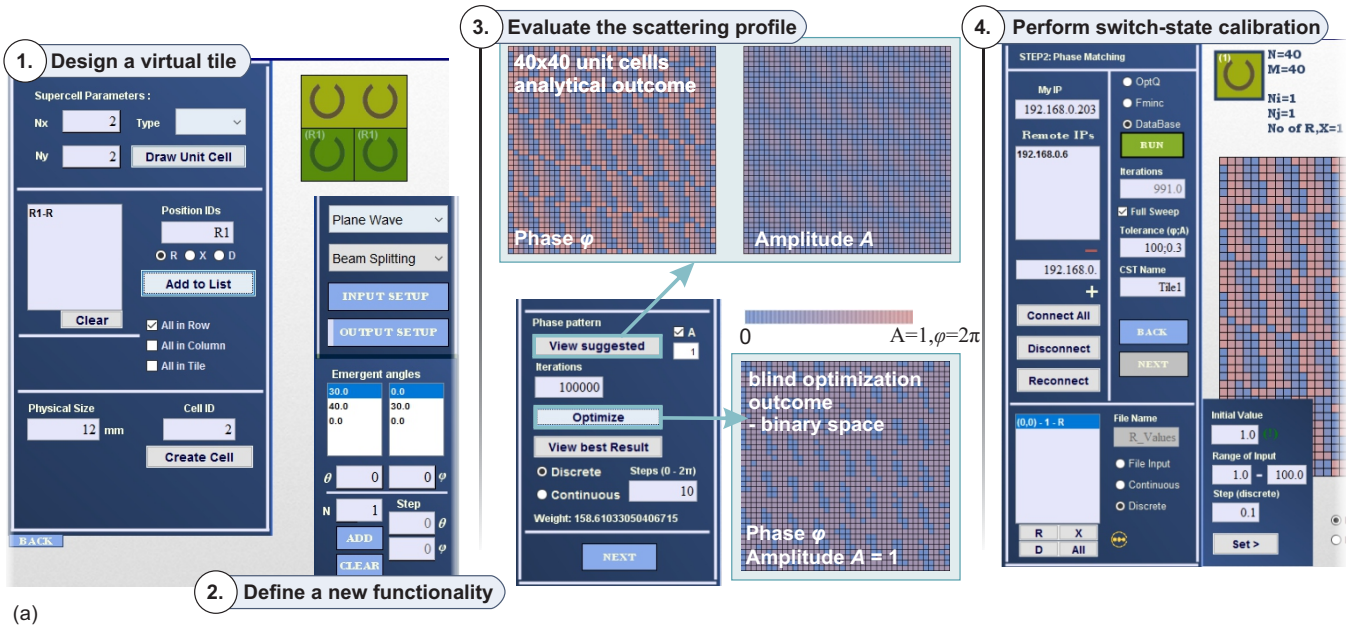
In the following, we list and comprehensively describe the steps undertaken during a optimization process, as seen through the GUI environment of the Metamaterial Middleware. In summary, this process involves:

- The definition of a new unit cell structure and tile array (if required).
- The parameterization of a new functionality.
- The analytical evaluation of the scattering profile on the metamaterial for the selected functionality.

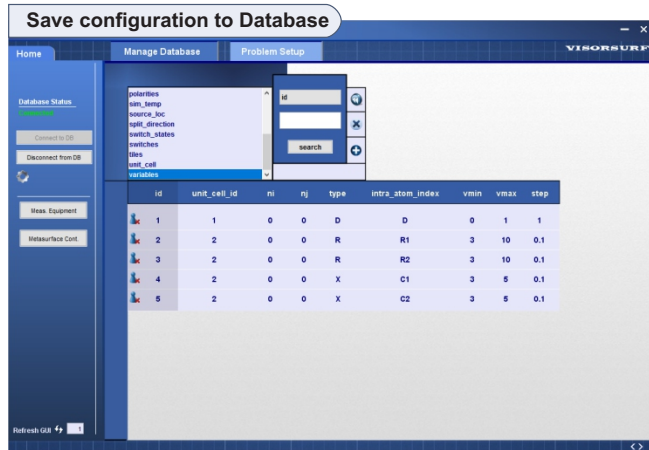
- The association of the metamaterial profile to the set of element states, through the use of numerical simulations.
- The experimental evaluation of the exported configuration through physical measurement of a metamaterial prototype. Notably, the presented software has been verified experimentally, and a full report can be found online [20].
- The final storing of all configuration parameters into the configuration DB to complete the function optimization process.

In this context, Fig. 11(a) depicts all the individual steps in separate panels. If a new configuration is to be defined for a tile already present in the configuration DB, then, the first step can be skipped. Alternatively, the user must input all essential parameters of the unit cell structure, i.e. the number and type of all variables that correspond to the sum of reconfigurable metamaterial elements. The definition of a new configuration begins with the parameterization of the desired functionality (Fig. 11(a.2)). The current implementation supports plane wave or point source inputs (for far- and near-field energy sources) and a set of output options corresponding to all basic metamaterial functionalities, discussed in Section 2. Here, we select a beam splitting operation and proceed to the first main step of the characterization process.

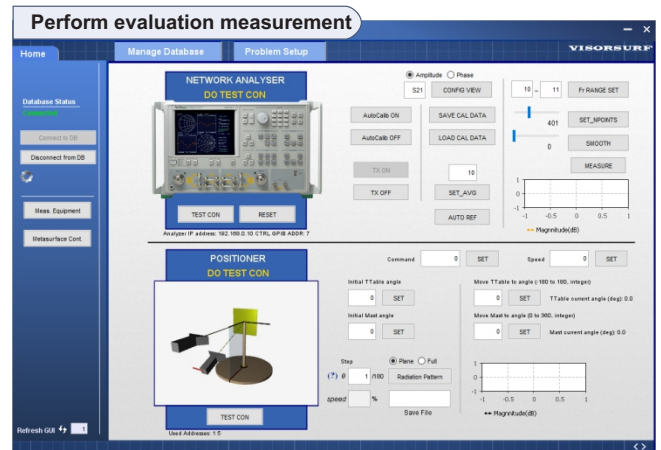
The analytical evaluation of the energy scattering profile is performed in the software locally and in real time. During this step, the Metamaterial Middleware calculates the proper scattered field response of the impinging wave, for each unit cell at the $N \times M$ tile, via either the analytical methods of Section 5 or through an optimization process. The scattered fields are evaluated for each unit cell as a double complex variable ($A_1 e^{i\phi_1}$ and $A_2 e^{i\phi_2}$, magnitude and phase of the reflected TE and TM polarizations, respectively), a process physically correct under the condition that the unit cell is a subwavelength entity. This simply implies that the input of the “physical size” field in Fig. 11(a.1) must comply with this specification or a warning message will appear. In our example, the selected tile consists of binary elements (D stands for diodes), which may only control the phase of the co-polarized scattered field (ϕ_1 term). By clicking the “View suggested” button, we analytically calculate and display $\phi_0 - \phi_1$ (ϕ_0 is the phase of the incoming wavefront), which should give an indication of the diode states at the metamaterial. Alternatively, a similar result can be extracted by launching the metaheuristic optimizer, either via a blind optimization process (all-0 initial solution) or an assisted optimization, using the analytically evaluated profile as an initial solution. The latter practice leads to more refined results, over an analytical evaluation, by considering the finite size of the tile and the non-infinitesimal size of the unit cell.



(a)



(b)



(c)



(d)

Fig. 11 – (a) Snapshots of the Metamaterial Middleware GUI during the metamaterial characterization process. The user may select an existing tile or create a new one by drawing its unit cell topology and variable elements within. A new functionality for this tile is assigned by selecting the desired parameters for the impinging and outgoing waves. The process begins by evaluating the scattering response profile, calculated as a phase and amplitude array for the selected metamaterial function. During the final step the Metamaterial Middleware seeks the set of optimal states that can realize the scattering response into an actual hardware configuration. (b) After a successful characterization process the configuration is saved to the DB. Herein, as an instance, a snapshot of the DB manager that was created during development of the software is presented. (c) The Metamaterial Middleware offers extensive experimental capabilities through a dedicated module. (d) Utilizing the hardware present in an anechoic chamber, the Metamaterial Middleware was able to acquire full scattering diagrams of a metamaterial tile, mounted on the available positioning equipment. The three-dimensional (3D) schematic, shown on the left, is updated live as the turntable or the positioner head rotates during a measurement or optimization process. On the right hand side, the Metamaterial Middleware host (white laptop) connects via Ethernet to the VNA (measuring the Rx/Tx antennas) and to the positioner controller.

In the final step, we seek to match the scattering field response (i.e. the $(A_1 e^{i\phi_1}, A_2 e^{i\phi_2})_{i,j}$ pairs calculated in the previous step) to the appropriate set of element states, such as the resistance R and reactance X (capacitance or inductance) of the loads, for all unit cells indexed by i, j . The correspondence between the scattering response of a unit cell and its physical structure constitutes a highly complex and demanding propagation problem. Actually, the search for a proper set of states for a fully defined response constitutes an inverse problem with closed-form solutions available only for very simple unit cell types. As such, it demands the use of highly efficient optimization algorithms and strong computation power to perform the necessary numerical simulations. In our implementation, we employ a cluster of interconnected computer clients (that serve as simulation nodes) receiving instructions through a TCP/IP network from the host Metamaterial Middleware running on the main machine. In a lightweight scenario, a single-cell simulation assigned to one of the clients can be completed in less than a minute but this may grow substantially when the complexity of the design is increased. An acceptable convergence is expected to be reached within a few thousands simulation trials. In perspective, the prototype studied in this work displayed a typical evaluation time of 8 to 12 hours for a full characterization of its configuration space and all possible impinging plane waves (θ, ϕ sets) at the operation frequency of the hardware. The evaluation is performed through Algorithm 1 (see next page) whose steps are described below. Specifically, the user:

- selects one of the three options: i) a gradient based optimizer, recommended for a small number of variables (less than four), ii) a metaheuristic optimizer, recommended for a higher count of variables (more than three), and iii) a database-based optimization process that configures the proper switch-states, through simulation results already stored in the configuration DB.
- suggests a convergence limit Tol to the optimizer in the Tolerance text box for all variable targets,
- fills the IP list with all available simulation nodes and initiates a connection. All nodes will launch the installed simulation software, open the corresponding geometry model, and remain idle until further instructions are received.
- modifies, if necessary, the value range, step and initial value for each variable in the unit cell.
- begins the optimization process by clicking the “RUN” button. The software will iterate over all unit cells (from top-left $(0,0)$ to bottom-right (N,M)), seeking a set of optimal values for the variables of each cell. During this sequence, each unit cell (i,j) initiates an independent optimization

Algorithm 1 Physical Element Calibration. **Highlighted commands** run in a separate thread.

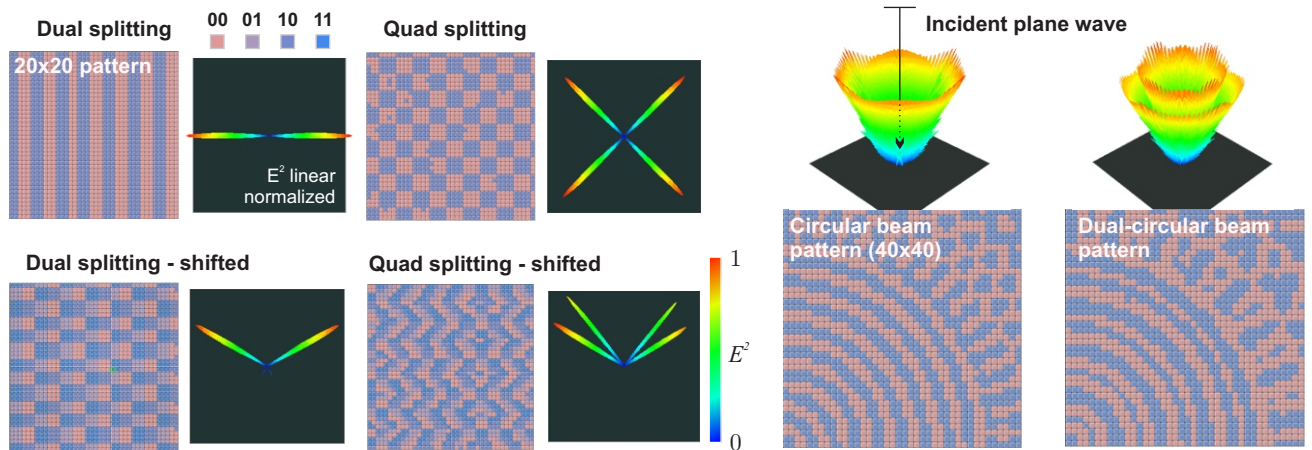
```

Set  $index$  to  $i = 0, j = 0$ 
while  $index$  does not exceed  $max$  do
  Get  $S_{ij}$ ; Get  $Tol$ 
  if  $\neg$ (exists  $S_{db}$  where  $|S_{db} - S_{ij}| < Tol$ ) then
    while searching do
      Select node where  $state == free$ 
      Set to  $busy$ 
      Initiate parallel thread;
      Optimizer: Fixes  $(R, L, C, D)$  variables;
      Optimizer  $\rightarrow$  Requests new simulation;
      node  $\leftarrow$  Returns  $S_{sim}$ 
       $S_{db} = S_{sim}$ ;
      if  $|S_{sim} - S_{ij}| < Tol$  then
         $searching = false$ 
        Set  $(R, L, C, D)$  variables as optimal
      end if
      Set node  $state$  to  $free$ 
    end while
    Iterate  $index$ 
  end if
end while

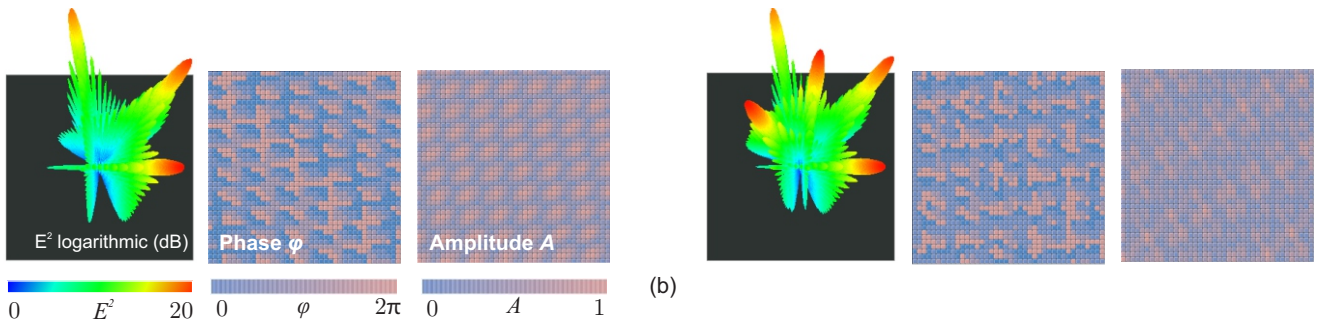
```

process and a new series of simulations begins until a sufficiently close convergence to the targeted scattering field response $S_{ij} = (A_1 e^{i\phi_1}, A_2 e^{i\phi_2})_{i,j}$ is achieved. Prior to each simulation, the optimizer looks up all entries in the associated table of the configuration DB, in case a result (S_{db}) is already stored. If not, the simulation will start and the result will be stored afterwards. Over time, this process can populate the DB with enough results, making option (iii) of the initial step a highly efficient method for evaluating new functionalities for this particular tile.

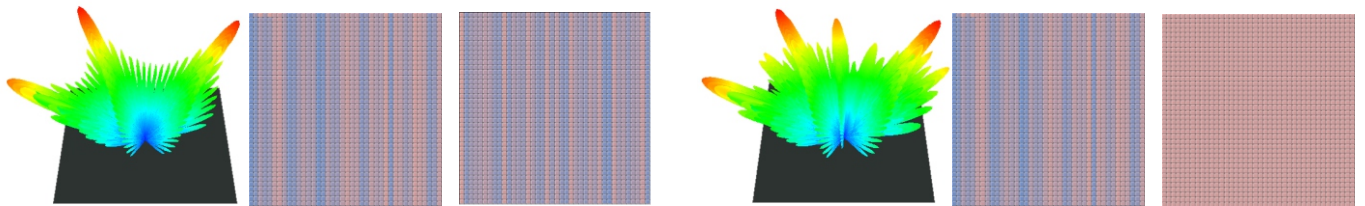
By completing the previous steps, the software has successfully defined a new configuration for the chosen tile, which, now, remains to be stored in the configuration DB. Before doing so, we can apply an additional evaluation step by conducting an experimental measurement on a physical prototype (if available). Our current implementation is able to assess multi-splitting or absorption functionalities by measuring the full scattering diagram on the front metamaterial hemisphere, which can then be compared to the scattering profile produced by the software. As presented in the corresponding technical report [20], several tests with a metamaterial unit have already been successfully conducted in a fully equipped anechoic chamber (Fig. 11). This test bed incorporates a variety of algorithms, meeting individual needs for accuracy and speed for various cases of scattering response measurements (e.g. a full 3D-pattern versus a 2D-slice might be required for arbitrary lobe scattering). A simple case is outlined in Algorithm 2, where the user has previously set the appropriate parameters in the GUI (Fig. 11(c)). The GUI automates



(a)



(b)



(c)

Fig. 12 – (a) Element states and far-field scattering diagram of a 4-bit metamaterial array for 6 different cases. (b) Element states and scattering diagrams for triple- and quintuple-beam scattering, assuming continuously adjustable states with absorption capabilities. (c) Comparison between a tile with resistor elements (left) and a tile without (right) for an in-plane triple-beam splitting functionality. A non-uniform amplitude pattern (A_{mn}) can eliminate all side-lobes and provide increased security to the signal. The incident field in all cases is a vertically impinging plane wave, chosen for simplicity; yet any plane wave or point source input can be considered by a corresponding shift of the phase for each unit cell.

Algorithm 2 Evaluate Scattering Response, 3D, slow rotation case. Highlighted commands refer to HW instructions.

```

Set  $N \leftarrow$  number of equidistant points on the hemisphere
Set  $P[N] \leftarrow$  Struct of  $(\theta, \phi)$  points
for all elements  $i$  in  $P$  do
    Send rotation command  $\rightarrow$  Positioner
    (Mast -  $P[i].\theta$ , Head -  $P[i].\phi$ , Speed)
    Receive  $\leftarrow$  Positioner feedback
    Refresh 3D Figure
    Send measurement command  $\rightarrow$  VNA
     $E[i] =$  Receive  $\leftarrow$  S21 power
    Refresh Plot graph
end for
Save  $(P[i], E[i])$  to DB
    
```

the process of measuring metamaterial devices in the anechoic chamber by supporting several communication interfaces for the following equipment:

- Vector network analyzer (VNA): produces the energy signal and receives the response (i.e. S_{21} -parameter) from the antenna setup.
- Positioner: allows the mechanical support of the metamaterial and antenna devices. Its controller can instruct the rotation of both heads (towards θ, ϕ), allowing a complete characterization of the scattering profile.
- metamaterial controllers: A metamaterial hosting reconfigurable elements incorporates a communication network for the explicit control of its element states. The Metamaterial Middleware implements the proper interface for the evaluation prototypes (serial port connection, WiFi, and Bluetooth have been integrated). The same interfaces are, also, used for the metamaterial API developed in Section 4.

A supplementary note is that the final switch-state configuration can be re-evaluated using the same metaheuristic optimizer utilized in step 3 of Fig. 12(a) via actual experimental results. The optimizer starts with the software-defined configuration as an initial solution and gradually adjusts the switch-state matrices to more optimally converge to the pursued functionality under true operational conditions. The implemented algorithm follows the template of Algorithm 2, where N correspond to the number or optimization variables (e.g. the number of scattering lobes) and a second *for* loop nests the existing loop, seeking to maximize the $Sum_i(E[i])$ metric.

For further evaluation purposes, we validate a number of indicative examples from the literature, based on previously measured and simulated results. Hence, Fig. 12(a) presents the optimization outcome for a 4-bit metamaterial array, which can switch over four available states per

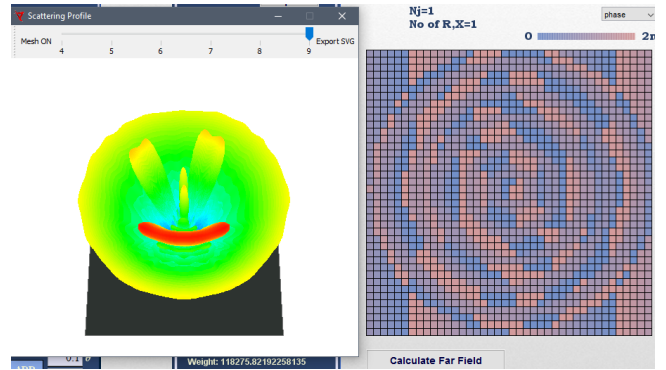


Fig. 13 – Arbitrary functionality optimization test. A smiley face-shaped scattering pattern is successfully produced (left). The corresponding metamaterial element configuration (different metamaterial states expressed in colormap) is shown to the left.

unit cell with reflection phases ($-90, 0, 90, 180$) and full reflection amplitude ($A = 1$) [14, 38]. The outcome complies fully with the results provided by the corresponding authors, indicating that the Metamaterial Middleware may cooperate with arbitrary hardware configurations and thus be compatible with any reasonable design in a future diverse metamaterial market. Following these results, we, also, test four exclusive cases that highlight the additional capabilities of our software. In particular, Fig. 12(b) displays the results for a triple- and a quintuple-beam splitting case, while Fig. 12(c) shows the optimization outcome for an in-plane triple-beam splitting functionality. For the latter case, the integrated theoretical algorithms were able to eliminate all side lobes by suggesting a non-uniform pattern for the amplitude A of the co-polarized scattered field. This particular case demands a tile with controllable absorption elements (resistors). Finally, in Fig. 13 we proceed to showcase the optimization of an arbitrary departing wavefront formation. A smiley-face shaped scattering pattern is selected as the required energy wave response of the metamaterial to a planar impinging wave. The optimization process successfully produces the required wavefront, and the corresponding metamaterial element states are shown to the right of Fig. 13.

7. DISCUSSION: THE TRANSFORMATIONAL POTENTIAL OF THE IOMMT AND FUTURE DIRECTIONS

While the potential of the IoMMT paradigm alone may be worth the investigation, here we evaluate its practical opportunities affecting the industry, the end users and the environment, namely:

- How can the IoMMT prolong the life cycle of products across deployment scales?
- How can the IoMMT help maintain a high-speed product development pace, without sacrificing ecological concerns during the product design phase?

To these ends, we believe that the concept of Circular Economy (CE) and its associated performance indices is a fitting framework for the initial exploration and evaluation of the IoMMT paradigm [39, 40]. CE seeks to make technological products reusable, repairable and recyclable across their lifetime (i.e., development, purchase, usage and disposal) by introducing cross-product and cross-manufacturer interactions. Instead of the traditional, linear order of life cycles phases, i.e., i) raw resource acquirement, ii) processing, iii) distribution, iv) its use and v) disposal, the CE advocates to create links from disposal to all preceding phases, promoting i) re-processing or refurbishment, ii) redeployment and redistribution, and iii) multiple uses.

However, according to the literature [41], the CE introduces a paradoxical tension in the industry: While the industry is pressed for faster growth and, hence, a faster product development rate, CE can introduce a series of design considerations that make for a slower product development rate. In this view, the paradigm of IoMMT is by its nature impactful for the energy and ecological footprint of multiple products, across disciplines and scales.

The fact that it enables the tuning and optimization of the physical properties of matter allows for a tremendous impact both in terms of quality of service per product and scale, but also for energy savings in a horizontal manner. Moreover, the IoMMT can contribute a software-driven way for optimizing material properties. Using this new technology, the industrial players can maintain a fast-paced product design, where energy efficiency and sustainability can be upgraded programmatically via “eco-firmware” during its use, thereby offloading the product design phase of such concerns.

An important future research direction of the IoMMT is to quantify the financial savings stemming from adopting this technique. While the Circular Economy-derived paradoxical tensions have been around for long and are hard to eradicate, IoMMT can facilitate their resolution by quantifying them, potentially aligning fast-paced marketing with environmental sustainability.

Apart from the CE line of work, future work will seek to provide the theoretical and modeling foundations of the IoMMT. Our vision, overviewed in Fig. 14, is for a full-stack study of this new concept, covering: the physical layer modeling, the internetworking and communications layer, and the application layer.

At the physical layer, future research needs to classify and model metamaterials in a functionality-centric way, and introduce fitting Key-Performance Indicators for each offered energy manipulation type. Metasurface-internal control variations (technologies and ways of monitoring embedded active elements) can be taken into account, and aspire to deduce models covering the aspects of control data traffic, energy expenditure and feature-based manufacturing cost estimation. This will enable the creation of the first, cross-physical domain profiling and recommendation system for metamateri-

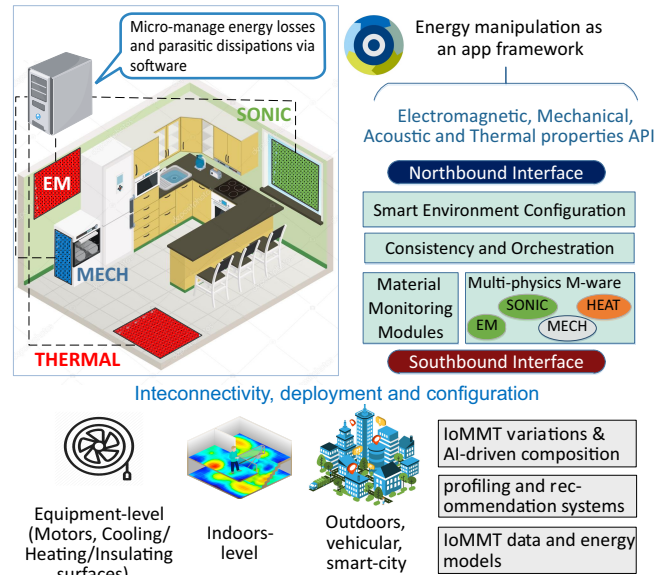


Fig. 14 – Envisioned future research directions for the Internet of MetaMaterials.

als, to match the requirements and specifications of any envisioned application.

Moving to the networking layer, research can follow the proposed northbound/southbound abstraction model inspired from the SDN paradigm. This will provide the necessary platform for: i) Interconnecting the IoMMT to the vast array of existing networked devices and assorted standards, models and protocols. ii) Provide the necessary software abstractions, to open the field of energy micromanagement to software developers (i.e., without specialty in Physics), enabling the energy-propagation-as-an-app paradigm. In this aspect, we also envision the need for algorithms optimizing IoMMT deployments for minimal-investment-maximal-control, and orchestrating IoMMT deployments for any set of generic energy micromanagement objective.

The control time granularity depends on the application scenario and the volatility of the factors affecting the energy propagation within an environment. In an indoors wireless communications setting, such as the one studied in [7], where the Intelligent Wireless Environment needs to continuously adapt to the position of user devices, the control granularity can be considered to be 10-25 msec (i.e., randomly walking users running a mobile application).

Finally, at the application layer research can define key-applications of the IoMMT at multiple scales, starting from the internals of equipment, such as motors, heating, cooling and insulating surfaces. This will provide the basic units for IoMMT incorporation to devices spanning home appliances (ovens, refrigerators, washers, heating and cooling units), electronics (from interference cancellation, to smart cooling) and building materials (acoustic, thermal and mechanical insulators). Various scales can be taken into consideration, from indoor (smart-house) and outdoor (city-level, ve-

hicular networks) IoMMT deployments.

8. CONCLUSION

In this paper, we pursue the systematic expansion of the concept of software-defined metamaterials by establishing the key software elements of a metamaterial network. With this goal in mind, we introduce two special categories of software, the Metamaterial Middleware, which can methodically produce novel configurations for a single metamaterial tile and the metamaterial API, which is in charge of supervising the energy propagation within a metamaterial-coated space. The two components were studied and developed autonomously, and then merged into a unified application via a common layer of abstraction.

For the API, we explored the means to interpret a metamaterial, its configuration space, and the supported energy manipulation functionalities via a group of well-defined software objects. The key objective was to successfully conceal the physical layer of the metamaterial, and only expose the essential parameters for configuring an operation. In light of this definition, the API is capable of instructing its environment to steer, focus, absorb or split the incoming signals through a simple interface, without any reference to the underlying physics.

The Metamaterial Middleware was developed by means of various theoretical and computational tools, including analytical algorithms that assess the scattering response, full-wave simulations that accurately evaluate a unit cell response, and experimental modules that can perform physical measurements of metamaterial devices. Through a step-by-step process we defined a robust characterization methodology, supporting a large set of metamaterial operations with no restriction on the specifications for the metamaterial. We also discuss the prospect of applying different optimization algorithms for each stage of the metamaterial characterization process. Finally, the evaluation of the middleware outcomes demonstrated that the IoMMT paradigm can be employed successfully within a unified framework.

ACKNOWLEDGMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme-project C4IIoT, GA EU833828. The authors also acknowledge FETOPEN-RIA project VISORSURF. All presented software modules were conceived, designed and implemented in full by the Foundation for Research and Technology-Hellas (FORTH) and G. Pyrialakos served as the lead developer.

REFERENCES

- [1] J. Pan, R. Jain, S. Paul, T. Vu, A. Saifullah, and M. Sha, "An internet of things framework for smart energy in buildings: Designs, prototype, and experiments," *IEEE Internet Things J.*, vol. 2, no. 6, pp. 527–537, Dec. 2015. [Online]. Available: <https://doi.org/10.1109/jiot.2015.2413397>
- [2] F. Capolino, *Theory and phenomena of metamaterials*. CRC press, 2017.
- [3] A. Li, S. Singh, and D. Sievenpiper, "Metasurfaces and their applications," *Nanophotonics*, vol. 7, no. 6, pp. 989–1011, 2018.
- [4] M. Kadic, T. Bückmann, R. Schittny, and M. Wegener, "Metamaterials beyond electromagnetism," *Reports on Progress in physics*, vol. 76, no. 12, p. 126501, 2013.
- [5] J. U. Surjadi, L. Gao, H. Du, X. Li, X. Xiong, N. X. Fang, and Y. Lu, "Mechanical metamaterials and their engineering applications," *Advanced Engineering Materials*, vol. 21, no. 3, p. 1800864, 2019.
- [6] M. Pishvar and R. L. Harne, "Foundations for soft, smart matter by active mechanical metamaterials," *Advanced Science*, p. 2001384, 2020.
- [7] C. Liaskos, A. Tsioliariidou, A. Pitsillides, S. Ioannidis, and I. Akyildiz, "Using any surface to realize a new paradigm for wireless communications," *Commun. ACM*, vol. 61, pp. 30–33, 2018.
- [8] C. Liaskos, A. Tsioliariidou, S. Nie, A. Pitsillides, S. Ioannidis, and I. Akyildiz, "On the network-layer modeling and configuration of programmable wireless environments," *IEEE/ACM Trans. Netw.*, vol. 27, no. 4, pp. 1696–1713, 2019.
- [9] I. F. Akyildiz and J. M. Jornet, "The internet of nano-things," *IEEE Wireless Communications*, vol. 17, no. 6, pp. 58–63, 2010.
- [10] S. B. Glybovski, S. A. Tretyakov, P. A. Belov, Y. S. Kivshar, and C. R. Simovski, "Metasurfaces: From microwaves to visible," *Phys. Rep.*, vol. 634, pp. 1–72, 2016.
- [11] H.-T. Chen, A. J. Taylor, and N. Yu, "A review of metasurfaces: Physics and applications," *Rep. Prog. Phys.*, vol. 79, no. 7, pp. 076 401(1–40), 2016.
- [12] C. Huang, C. Zhang, J. Yang, B. Sun, B. Zhao, and X. Luo, "Reconfigurable metasurface for multifunctional control of electromagnetic waves," *Adv. Opt. Mater.*, vol. 5, no. 22, pp. 1 700 485(1–6), 2017.
- [13] Ö. Özdogan, E. Björnson, and E. Larsson, "Intelligent reflecting surfaces: Physics, propagation, and pathloss modeling," *IEEE Wireless Commun. Lett.*, vol. 9, no. 5, pp. 581–585, 2019.
- [14] Q. Zhang, X. Wan, S. Liu, J. Yin, L. Zhang, and T. Cui, "Shaping electromagnetic waves using software-automatically-designed metasurfaces," *Sci. Rep.*, vol. 7, no. 1, pp. 3588(1–11), 2017.

- [15] A. P. Mosk, A. Lagendijk, G. Lerosey, and M. Fink, "Controlling waves in space and time for imaging and focusing in complex media," *Nature photonics*, vol. 6, no. 5, pp. 283–292, 2012.
- [16] X. Tan, Z. Sun, D. Koutsonikolas, and J. M. Jornet, "Enabling indoor mobile millimeter-wave networks based on smart reflect-arrays," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 270–278.
- [17] C. Liaskos, A. Tsioliariidou, A. Pitsillides, I. F. Akyildiz, N. V. Kantartzis, A. X. Lalas, X. Dimitropoulos, S. Ioannidis, M. Kafesaki, and C. Soukoulis, "Design and development of software defined metamaterials for nanonetworks," *IEEE Circuits and Systems Magazine*, vol. 15, no. 4, pp. 12–25, 2015.
- [18] F. Liu *et al.*, "Programmable metasurfaces: State of the art and prospects," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 2018-May. IEEE, May 2018, p. 8351817.
- [19] T. Saeed, V. Soteriou, C. Liaskos, A. Pitsillides, and M. Lestas, "Toward fault-tolerant deadlock-free routing in hypersurface-embedded controller networks," *IEEE Networking Letters*, vol. 2, no. 3, pp. 140–144, 2020.
- [20] The VISORSURF project consortium, "A hypervisor for metasurface functionalities: progress report," *European Commission Project VISORSURF: Public Report, Aug-2019*, [Online:] <https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5c560b376&appId=PPGMS>.
- [21] T. Frenzel, M. Kadic, and M. Wegener, "Three-dimensional mechanical metamaterials with a twist," *Science*, vol. 358, no. 6366, pp. 1072–1074, 2017.
- [22] M. R. Haberman and M. D. Guild, "Acoustic metamaterials," *Physics Today*, vol. 69, no. 6, pp. 42–48, Jun. 2016. [Online]. Available: <https://doi.org/10.1063/pt.3.3198>
- [23] M. S. Dresselhaus, G. Chen, M. Y. Tang, R. Yang, H. Lee, D. Wang, Z. Ren, J.-P. Fleurial, and P. Gogna, "New directions for low-dimensional thermoelectric materials," *Advanced materials*, vol. 19, no. 8, pp. 1043–1053, 2007.
- [24] C. Liaskos, S. Nie, A. Tsioliariidou, A. Pitsillides, S. Ioannidis, and I. F. Akyildiz, "Mobility-aware beam steering in metasurface-based programmable wireless environments," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 9150–9154.
- [25] C. Liaskos, S. Nie, A. Tsioliariidou, A. Pitsillides, S. Ioannidis, and I. Akyildiz, "End-to-end wireless path deployment with intelligent surfaces using interpretable neural networks," *IEEE Transactions on Communications*, 2020.
- [26] F. Mathioudakis, C. Liaskos, A. Tsioliariidou, S. Nie, A. Pitsillides, S. Ioannidis, and I. Akyildiz, "Advanced physical-layer security as an app in programmable wireless environments," in *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2020, pp. 1–5.
- [27] Z. Chen, B. Guo, Y. Yang, and C. Cheng, "Metamaterials-based enhanced energy harvesting: A review," *Physica B: Condensed Matter*, vol. 438, pp. 1–8, 2014.
- [28] B. Florijn, C. Coulais, and M. van Hecke, "Programmable mechanical metamaterials," *Phys. Rev. Lett.*, vol. 113, no. 17, pp. 175503(1–4), 2014.
- [29] S. A. Cummer, J. Christensen, and A. Alù, "Controlling sound with acoustic metamaterials," *Nat. Rev. Mat.*, vol. 1, no. 3, pp. 16001(1–13), 2016.
- [30] L. Zhang, X. Q. Chen, S. Liu, Q. Zhang, J. Zhao, J. Y. Dai, G. D. Bai, X. Wan, Q. Cheng, G. Castaldi, V. Galdi, and T. J. Cui, "Space-time-coding digital metasurfaces," *Nat. Commun.*, vol. 9, no. 1, pp. 4334(1–11), 2018.
- [31] Y. E. Oktian, S. Lee, H. Lee, and J. Lam, "Distributed SDN controller system: A survey on design choice," *Computer Netw.*, vol. 121, pp. 100–111, 2017.
- [32] C. Liaskos, A. Pitilakis, A. Tsioliariidou, N. Kantartzis, and A. Pitsillides, "Initial UML definition of the hypersurface compiler middleware," *European Commission Project VISORSURF: Public Deliverable D2.2, 31-Dec-2017*, [Online:] <http://www.visorsurf.eu/m/VISORSURF-D2.2.pdf>.
- [33] N. Ashraf, M. Lestas, T. Saeed, H. Taghvaei, S. Abadal, A. Pitsillides, and C. Liaskos, "Extremum seeking control for beam steering using hypersurfaces," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2020, pp. 1–6.
- [34] M. Laguna and R. Marti, "The OptQuest callable library," in *Optimization Software Class Libraries*. Springer, 2003, pp. 193–218.
- [35] C. Liaskos, A. Tsioliariidou, S. Nie, A. Pitsillides, S. Ioannidis, and I. Akyildiz, "An interpretable neural network for configuring programmable wireless

environments,” in *IEEE Int. Workshop Signal Processing Adv. Wireless Commun. (SPAWC 2019)*, 2019, pp. 1–5.

- [36] H. Taghvaei, A. Jain, X. Timoneda, C. Liaskos, S. Abadal, E. Alarcón, and A. Cabellos-Aparicio, “Radiation pattern prediction for metasurfaces: A neural network based approach,” *arXiv preprint arXiv:2007.08035*, 2020.
- [37] Y. Saifullah, A. B. Waqas, G.-M. Yang, F. Zhang, and F. Xu, “4-bit optimized coding metasurface for wideband rcs reduction,” *IEEE Access*, vol. 7, pp. 122 378–122 386, 2019.
- [38] S. Liu, T. J. Cui, L. Zhang, Q. Xu, W. Qiu, X. Wan, J. Q. Gu, W. X. Tang, M. Q. Qi, J. G. Han, W. L. Zhang, X. Y. Zhou, and Q. Cheng, “Convolution operations on coding metasurface to reach flexible and continuous controls of terahertz beams,” *Adv. Science*, vol. 3, no. 10, pp. 1 600 156(1–12), Jul. 2016.
- [39] W. R. Stahel, “The circular economy,” *Nature*, vol. 531, no. 7595, pp. 435–438, 2016.
- [40] C. Liaskos, A. Tsioliariidou, and S. Ioannidis, “Towards a circular economy via intelligent metamaterials,” in *2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*. IEEE, 2018, pp. 1–6.
- [41] T. Daddi, D. Ceglia, G. Bianchi, and M. D. de Barcellos, “Paradoxical tensions and corporate sustainability: A focus on circular economy business cases,” *Corporate Social Responsibility and Environmental Management*, vol. 26, no. 4, pp. 770–780, 2019.

AUTHORS



Christos Liaskos received a Diploma in Electrical and Computer Engineering from the Aristotle University of Thessaloniki (AUTH), Greece in 2004, an MSc degree in Medical Informatics in 2008 from the Medical School, AUTH and PhD degree in Computer Networking from the Dept. of Informatics, AUTH in 2014. He has published work in several journals and conferences, such as *IEEE Transactions on: Networking, Computers, Vehicular Technology, Broadcasting, Systems Man and Cybernetics, Networks and Service Management, Communications, INFOCOM*. He is currently an assistant professor with the University of Ioannina, Greece, and an affiliated researcher at the Foundation of Research and Technology, Hellas (FORTH). His research interests include computer networks, security and nanotechnology, with a focus on developing nanonetwork architectures and communication protocols for future applications.



George Pyrialakos received his Diploma and Ph.D. degree from the Dept. of Electrical and Computer Engineering, Aristotle University of Thessaloniki (AUTH), Greece, in 2013 and 2019, respectively. He is currently a post-doc researcher in AUTH, affiliated with the Foundation for Research and Technology Hellas (FORTH). His research interests include computational electromagnetics, metamaterials, photonics, and condensed matter physics.



Alexandros Pitilakis received his Diploma and Ph.D. degree from the Dept. of Electrical and Computer En-

gineering, Aristotle University of Thessaloniki (AUTH), Greece, in 2005 and 2013. He, also, holds an MSc. degree from the ENST Paris, 2007. He is a post-doc researcher in AUTH, affiliated with the Foundation for Research and Technology Hellas (FORTH). His research interests include computational electromagnetics, metamaterials, and nonlinear optics.



Nikolaos Kantartzis is a professor at the School of Electrical and Computer Engineering, Aristotle University of Thessaloniki (AUTH), Greece, from where he received the Diploma and Ph.D. degrees in 1994 and 1999, respectively. His primary research interests include computational electromagnetics, metamaterials, graphene, EMC/EMI problems, microwaves, and nano devices. Dr. Kantartzis is a Senior Member of IEEE, an ICS and ACES member.



Ageliki Tsioliaridou received a Diploma and PhD degrees in Electrical and Computer Engineering from the Democritus University of Thrace (DUTH), Greece, in 2004 and 2010, respectively. Her research work is mainly in the field of Quality of Service in computer networks. Additionally, her recent research interests lie in the area of nanonetworks, with a specific focus on architecture, protocols, security and authorization issues. She has contributed to a number of EU, ESA and National research projects. She is currently a researcher at the Foundation of Research and Technology, Hellas (FORTH).



Sotiris Ioannidis received a BSc degree in Mathematics and an MSc degree in Computer Science from the University of Crete in 1994 and 1996 respectively. In 1998 he received an MSc degree in Computer Science from the University of Rochester and in 2005 he received his PhD from the University of Pennsylvania. Ioannidis held a Research Scholar position at the Stevens Institute of Technology until 2007, and since then he is Research Director at the Institute of Computer Science of the Foundation for Research and Technology - Hellas. Since November 2017 he is a member of the European Union Agency for Network and Information Security (ENISA) Permanent Stakeholders Group (PSG). His research interests are in the area of systems, networks, and security. Ioannidis has authored more than 100 publications in international conferences and journals, as well as book chapters, including ACM CCS, ACM/IEEE ToN, USENIX ATC, NDSS, and has both chaired and served in numerous program committees in prestigious international conferences. Ioannidis is a Marie-Curie Fellow and has participated in numerous international and European projects. He has coordinated a number of European and National projects (e.g. PASS, EU-INCOOP, GANDALF, SHARCS) and is currently the project coordinator of the THREAT-ARREST, I-BiDaaS, BIO-PHOENIX, IDEAL-CITIES, CYBERSURE, and CERTCOOP European projects.



Michail Christodoulou received his Diploma and Ph.D. degree in Electrical and Computer Engineering from the Dept. of Electrical and Computer Engineering, Aristotle University of Thessaloniki (AUTH) in 1994 and 2006, respectively. In 1995, he received an MSc. degree from the University of Bradford, UK. In 2014, he joined AUTH as a Postdoctoral Research Fellow. His research interests include RF circuits, antennas, RF MEMS, and electromagnetic measurements.



Andreas Pitsillides is a Professor in the Department of Computer Science, University of Cyprus, heads NetRL, the Networks Research Laboratory he founded in 2002, and is appointed Visiting Professor at the University of the Witwatersrand (Wits), School of Electrical and Information engineering, Johannesburg, South Africa. Earlier (2014-2017) Andreas was appointed Visiting Professor at the University of Johannesburg, Department of Electrical and Electronic Engineering Science, South Africa. His broad research interests include communication networks (fixed and mobile/wireless), Nanonetworks and Software Defined Metasurfaces/Metamaterials, the Internet- and Web- of Things, Smart Spaces (Home, Grid, City), and Internet technologies and their application in Mobile e-Services, especially e-health, and security. He has a particular interest in adapting tools from various fields of applied mathematics such as adaptive non-linear control theory, computational intelligence, game theory, and recently complex systems and nature inspired techniques, to solve problems in communication networks. He has published over 270 referred papers in flagship journals (e.g. IEEE, Elsevier, IFAC, Springer), international conferences and book chapters, 2 books (one edited), participated in over 30 European Commission and locally funded research projects as principal or co-principal investigator, received several awards, including best paper, presented keynotes, invited lectures at major research organisations, short courses at international conferences and short courses to industry, and serves/served on several journal and conference executive committees.



Ian F. Akyildiz received his BS, MS, and PhD degrees in Electrical and Computer Engineering from the University of Erlangen-Nürnberg, Germany, in 1978, 1981 and 1984, respectively. Currently he serves as a Consulting Chair Professor with the Computer Science Department at the University of Cyprus since January 2017.

He is also the Megagrant Leader with the Institute for Information Transmission Problems at the Russian

Academy of Sciences, in Moscow, Russia, since May 2018. He serves on the Advisory Board for the newly established research center called Technology Innovation Institute (TII) in Abu Dhabi, United Arab Emirates since June 2020. He is the President of the Truva Inc. since March 1989 and Scientific Advisor for the newly established company Airanaculus since April 2020. He is a Visiting Distinguished Professor with the SSN College of Engineering in Chennai, India since October 2019. Dr. Akyildiz is an Adjunct Professor with the Department of Electrical and Computer Engineering at the University of Iceland since September 2020. He is the Ken Byers Chair Professor Emeritus in Telecommunications at the Georgia Institute of Technology, and the Director of the Broadband Wireless Networking Laboratory and Former Chair of the Telecom Group from 1985-2020. He is also former Finnish Distinguished Professor with the University of Tampere, Finland, supported by the Finnish Academy of Sciences from 2012-2016. He is the founder of NanoNetworking Center and Former Honorary Professor at the University of Politecnica de Catalunya in Barcelona from 2008-2017. Dr. Akyildiz is also former Distinguished Professor and Founder of the Advanced Wireless Networks Lab with the King Abdulaziz University in Jeddah, Saudi Arabia from 2011-2018. He is also the Founder of the Advanced Wireless Sensor Networks lab and former ExtraOrdinary Professor with the University of Pretoria, South Africa from 2007-2012.

He is the Founder and Editor in Chief of the newly established ITU J-FET (International Telecommunication Union Journal for Future and Evolving Technologies since July 2020). Dr. Akyildiz is Editor-in-Chief Emeritus of Computer Networks Journal (Elsevier) (1999-2019), the founding Editor-in-Chief Emeritus of the Ad Hoc Networks Journal (Elsevier) (2003-2019), the founding Editor-in-Chief Emeritus of the Physical Communication (PHYCOM) Journal (Elsevier) (2008-2017), and the founding Editor-in-Chief Emeritus of the Nano Communication Networks (NANOCOMNET) Journal (Elsevier) (2010-2017). Dr. Akyildiz launched many IEEE and ACM conferences. He is an IEEE Fellow (1995) and ACM Fellow (1996) and received numerous awards from IEEE and ACM and other professional organizations. His current research interests are in 6G Wireless Systems, Reconfigurable Intelligent Surfaces, Terahertz Communications, Nanonetworks, Internet of xThings (x=Underwater, Underground, Space/CubeSats, Nano and BioNano). He graduated 45 PhD students and 30 of them are in academia in very prestigious academic positions worldwide. He advised 13 Postdoctoral researchers. According to Google Scholar as of September 2020, his H-index is 124 and the total number of citations to his papers is 119+K. His rank in terms of h-index in the world is 46 and in the USA 32.