Cooperative Semantic Sensor Networks for pervasive computing contexts

Michele Ruta, Floriano Scioscia, Agnese Pinto, Filippo Gramegna, Saverio Ieva,
Giuseppe Loseto, Eugenio Di Sciascio
Politecnico di Bari - via E. Orabona 4, Bari (I-70125), Italy
{name.surname}@poliba.it

Abstract—**The Semantic Web of Things (SWoT) merges the Internet of Things with knowledge representation and reasoning techniques borrowed from the Semantic Web, in order to improve resource management and discovery. This paper proposes a SWoT framework in Wireless Sensor Networks (WSNs) enabling cooperative discovery of sensors and actuators. A backward-compatible extension of the Constrained Application Protocol (CoAP) makes possible to use semantic matchmaking via non-standard reasoning to better characterize the resource discovery. The framework also integrates nimble data stream mining to detect and annotate high-level events through raw data gathered from the environment. A cooperative environmental monitoring case study in Hybrid Sensor and Vehicular Networks (HSVN) is presented together with experiments on a real testbed to assess feasibility and benefits of proposal.**

Index Terms*—*CoAP, Cooperative sensing, Semantic Web, Internet of Things, Resource discovery, Matchmaking, Data mining

I. INTRODUCTION

The convergence of the Semantic Web and the Internet of Things (IoT) visions is known as the *Semantic Web of Things* (SWoT) [1]: semantically rich and machine-understandable information are envisioned to be embedded into real-world things, places and events, by means of inexpensive, unobtrusive, disposable micro-devices. This enables novel classes of advanced and interoperable services and applications. In order to implement SWoT frameworks, pervasive computing issues must be faced, including: volatility of resources and devices; hardware and software heterogeneity; adaptivity to context; computational resource limitations. In such a situation, resource discovery is a crucial feature.

The *Constrained Application Protocol* (CoAP) [2] is one of the most adopted application-layer protocols for interoperable object and sensor networks, but it currently supports only basic data-oriented representation and retrieval of resources, relying on string matching of attributes, with trivial “yes/no” results. Exact matches are quite unlikely in WSN scenarios, populated with heterogeneous sensors and devices from several independent vendors and administrators.

This paper proposes a comprehensive SWoT framework inducing decentralized, advanced discovery capabilities in Wireless Sensor Networks. It exploits semantic metadata to annotate data streams and sources, devices, events of interest and services, with a well-defined high-level meaning w.r.t. shared domain vocabularies (*ontologies*). The proposal includes: (i) backward-compatible slight extensions to CoAP and its resource discovery protocol, the *Constrained RESTful Environments (CoRE) Link Format*[1]; (ii) a semantic-based micro-matchmaking framework grounded on non-standard reasoning [3] featuring resource retrieval and ranking and supporting approximate matches; (iii) event detection and annotation by means of agile data mining algorithms on raw data gathered by a Semantic Sensor Network (SSN, *i.e.*, a semantic-enhanced WSN), using the SSN-XG ontology [4] as terminology. The proposed approach is clarified and further validated by means of a case study on cooperative environmental risk monitoring and management in Hybrid Sensor and Vehicular Networks (HSVNs). In order to execute experiments, the framework was implemented in a prototypical testbed with real devices.

The remainder of the paper is as follows. Related work is discussed in Section II, while Section III describes the proposed framework in detail. Section IV presents a possible case study, whose numerical experiments are reported in Section V, before conclusion.

II. RELATED WORK

CoAP [2] is an HTTP-like protocol for interconnected objects, designed for machine-to-machine interoperation in resource-constrained nodes. Following the REST (REpresentational State Transfer) architectural style, CoAP adopts a loosely coupled client/server model, based on stateless operations on *resource* representations. Clients access resources via synchronous request/response interactions, using HTTP-derived methods (`GET`, `PUT`, `POST`, `DELETE`). CoAP is acquiring relevance for its lightweight impact on storage and computation, resulting useful for a variety of application domains [5], [6], [7], and particularly Sensor Networks.

In latest years, interesting Semantic Sensor Networks (SSN) approaches were developed to integrate WSNs and smart objects with Semantic Web technologies. Every information resource in the Semantic Web is annotated with metadata in RDF[2] expressed w.r.t. an RDF Schema[3] or OWL[4] ontology.

---

[1] CoRE Link Format, IETF CoRE Working Group RFC 6690, August 2012, http://tools.ietf.org/html/rfc6690

[2] Resource Description Framework 1.1 Concepts and Abstract Syntax, W3C Recommendation, 25 February 2014. https://www.w3.org/TR/rdf11-concepts/

[3] RDF Schema 1.1, W3C Recommendation, 25 February 2014 https://www.w3.org/TR/rdf-schema/

[4] OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation, 11 December 2012, http://www.w3.org/TR/owl2-overview/

Language specifications include a standard XML serialization syntax. The adopted knowledge representation models are grounded on formal, logic-based semantics. Query languages, *e.g.*, SPARQL[5], are defined to extract and combine asserted information, while reasoning engines can automatically infer knowledge entailed by a given *Knowledge Base* (KB). Proposed solutions annotate data, devices and services w.r.t. ontologies and publish annotations as Linked Open Data (LOD) through RESTful Web services or interfaces compliant with Sensor Web Enablement (SWE) standards [8] of the Open Geospatial Consortium (OGC). OntoSensor [9] and SSN-XG [4] are among the best and most adopted ontologies for WSNs. The *SPITFIRE* [10] project introduced an infrastructure for semantic applications based on Internet-connected sensors and lightweight protocols like CoAP. Likewise, the ontology-based architecture in [11] exploited semantic annotations of sensor data for interoperability among different IoT technologies and to obtain higher-level knowledge from low-level sensor data. The *Sense2Web* [12] and *Linked Stream Middleware* (LSM) [13] platforms published sensor data as LOD on the Semantic Web, whereas in [14] semantic technologies were used to automatically orchestrate sensors. The system was able to combine sensors and services to satisfy user goals by means of semantic descriptions of functional and non-functional sensor properties. A more flexible approach is in [15], combining quantitative constraints and semantic reasoning to satisfy user requests expressed in terms of device attributes. As highlighted in Table I, most of the above works allowed only basic queries in SPARQL fragments on RDF annotations. More advanced resource discovery was not supported. Data and sensor management in mobile and pervasive contexts require techniques such as ontology-based *complex event processing* [16] and *semantic matchmaking* [3]. The latter in particular supports approximated matches and resource ranking with explanation of outcomes, by means of logic-based inference services.

III. SEMANTIC SENSOR NETWORK DISCOVERY FRAMEWORK

Resource discovery in basic CoAP exploits the *CoRE Link Format* specification. This protocol only enables a syntactic match of attributes, without a characterization of resource semantics. More sophisticated discovery is possible and needed in WSNs given more and more powerful and disposable off-the-shelf devices and more demanding applications.

In the approach proposed here, the semantic-based CoAP protocol enhancements of [17] were exploited to enable non-standard inferences devoted to an automated sensor discovery and composition. Semantic-based requests are similar to the standard ones, only including a novel usage of standard `URI-query` CoAP option along with novel resource attributes: `ro` (reference ontology), `sd` (request semantic description, compressed to reduce data transfers), `at` (annotation-type), `st` (semantic task), `sr` (semantic threshold). Context-dependent parameters can be also specified in a request through the following attributes: longitude (`lg`) and latitude (`lt`) of a reference geographical location (expressed in degrees) and maximum distance (`md`, in meters), in order to adopt a (center, distance) constraint allowing the server to pre-filter resources outside the requested area.

In cooperative sensor networks scenarios, data are usually gathered from several types of sensors to infer proper events. Hence the *Concept Covering* [3] inference results particularly useful when a CoAP client queries a SSN gateway to find the set of most suitable sensors, among those managed by sinks directly connected to the gateway. Each client can send a semantic-based request R expressed in OWL w.r.t. a shared ontology. The gateway carries out semantic matchmaking by solving a Concept Covering Problem (CCoP), in order to identify the set of resources which collectively satisfy the request to the best extent. In case of a partial cover, the outcome also includes: (i) the semantic description of the uncovered part (H) of the request; (ii) the percentage of request still not covered, obtained comparing H w.r.t. the original R (see [3] for algorithmic details). In case, a CoAP gateway has the possibility to forward the uncovered part as a new request towards other SSN nodes in the area of interest. In this way, each semantic-enabled gateway searches for more resources to satisfy lacking features through cooperative multi-hop discovery. The following subsections report on both system architecture and context mining details.

A. Architecture

Figure 1 shows the reference framework architecture. Sensors deployed in an area communicate with a local sink node, which acts as cluster head. A gateway is connected with multiple sinks, interfacing the network toward the outside. Each sensor is characterized by a semantic annotation describing its features and functionalities. Sink nodes embed CoAP servers able to register sensors along with their semantic annotations as CoAP resources and to support logic-based resource discovery operated by a lightweight embedded matchmaker [3]. They also gather and process data for event detection. When an event is recognized, it is annotated and a resource record is updated in the server. Each record also contains extra-logical context parameters, such as geographic coordinates and a timestamp. The gateway waits for resource discovery requests coming from client applications searching for events in the area, answering on behalf of connected sink nodes.

A modified version of *Californium* CoAP library [18] was used to implement the communication in Semantic Sensor Networks (SSNs), providing the semantic-based enhancements detailed in [17]. Two prototypical modules were developed to support the sensing process.

1. JOSM SSN Dashboard. A SSN dashboard was developed as a plugin for the Java OpenStreetMap (JOSM) open source editor[6], providing the following features:

---

[5] SPARQL Query Language for RDF, W3C Recommendation 15 January 2008, http://www.w3.org/TR/rdf-sparql-query/

[6] http://josm.openstreetmap.de/

TABLE I
COMPARISON OF THE PROPOSED APPROACH WITH THE STATE OF THE ART

Approach	**Application protocol**	**Representation language**	**Contextual query attributes**	**Distributed search**	**Match types**	**Resource ranking**	**Resource composition**
Pfisterer *et al.* [10]	CoAP	RDF	In SPARQL query	No	Exact only	No	No
Desai *et al.* [11]	CoAP, MQTT	RDF	–	No	Exact only	No	No
Barnaghi *et al.* [12]	HTPP	RDF	In SPARQL query	No	Exact only	No	Mashup composer
Le-Phuoc *et al.* [13]	HTPP	RDF	In SPARQL query	No	Exact only	No	Mashup composer
Tran *et al.* [14]	Agnostic	OWL 1.1	In OWL annotation	No	Exact and approximated	Yes (semantic distance)	Yes
Perera *et al.* [15]	Agnostic	RDF	In SPARQL query	Yes	Exact and approximated	Yes (Top-K on weighted attributes)	No
Taylor *et al.* [16]	Agnostic	OWL 2	In CEP language	No	Exact only	No	Yes
Proposed approach	CoAP	OWL 2	In CoAP parameters	Yes	Exact and approximated	Yes (semantic distance)	Yes

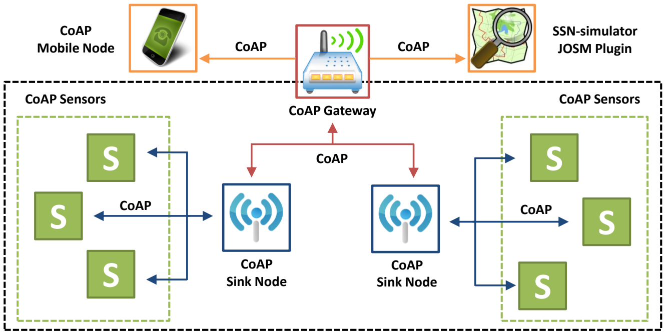
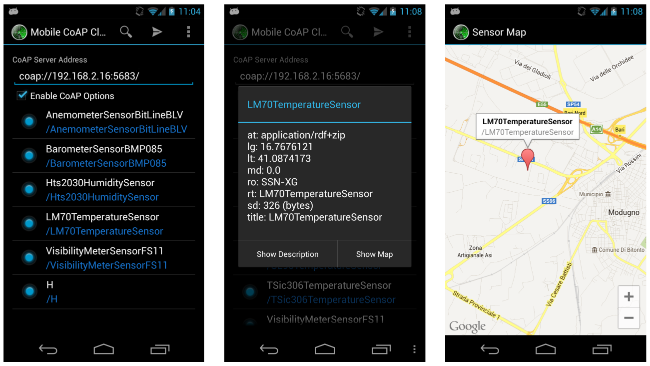


Fig. 1. CoAP-based Sensor Network Architecture

TABLE II
PARAMETERS FOR SCENARIO GENERATION

Parameter	**Description**
S	# of sink nodes
D_{min}	min # of CoAP sensors per sink
D_{max}	max # of CoAP sensors per sink
$dMax_S$	max distance in m between sink and sensors
G	# of CoAP gateways (GWs)
S_{min}	min # of sinks connected to a CoAP GW
S_{max}	max # of sinks connected to a CoAP GW
$dMax_G$	max distance in m between two connected GWs

– *SSN browsing*: it shows available sensors and sink nodes, registered on CoAP gateways;
– *Semantic-based sensor discovery*: it allows specifying a semantic-based CoAP request by means of the following features: reference location, maximum discovery range, inference task to perform and relevance threshold (minimum acceptable request covering level).
– *Scenario generation*. It integrates a module (extended from *OSM to Rescue*[7] [19]) to create random SSN configurations for large-scale simulation. Parameters for scenario customization are reported in Table II.
2. *CoAP Mobile Node*. An Android tool[8] was developed for to provide the following functionalities.
– *SSN browsing and sensor discovery*: each gateway node can be explored, to view all connected sensors or to retrieve the semantic-based discovery result set, as shown in Figure 2(a). Selecting a sensor, it is also possible to see its details as in Figure 2(b) and its location on the map (see Figure 2(c)).
– *Cooperative sensing*: the mobile node can be also configured automatically as an information source, by registering to the gateway temporarily. In such case, data from embedded micro-devices (*e.g.*, accelerometer, gyroscope) and wired or wireless connected sensing devices are used to enrich the reference environment annotation, in order to improve event detection. Mobile nodes are encouraged to share their knowledge with the rest of the SSN in order to get a better feedback.



coap://192.168.2.16:5683/

☑ Enable CoAP Options

AnemometerSensorBitLineBLV
/AnemometerSensorBitLineBLV
BarometerSensorBMP085
/BarometerSensorBMP085
Hts2030HumiditySensor
/Hts2030HumiditySensor
LM70TemperatureSensor
/LM70TemperatureSensor
VisibilityMeterSensorFS11
/VisibilityMeterSensorFS11
H
/H

(a) Discovery results

coap://192.168.2.16:5683/

LM70TemperatureSensor

at: application/rdf+zip
lg: 16.7676121
lt: 41.0874173
md: 0.0
ro: SSN-XG
rt: LM70TemperatureSensor
sd: 326 (bytes)
title: LM70TemperatureSensor

TSic306TemperatureSensor
/TSic306TemperatureSensor
VisibilityMeterSensorFS11

(b) Sensor details

LM70TemperatureSensor
/LM70TemperatureSensor

(c) Sensor location

Fig. 2. Mobile node

B. Data mining and context annotation

In WSN scenarios, large amounts of data have to be collected and interpreted to extract useful application information. Scenarios like environmental monitoring strongly require automatic procedures. The proposed framework includes a simple effective data mining method for SSN, designed to extract significant information on sensor readings and annotate it. The steps for identification of high-level events from sensory data streams are outlined hereafter:
– Both data streams from sensors embedded in the device and those from field sensors are read and collected in a given time window through standard CoAP requests. A list of elements is built, consisting of $< ID; value; timestamp >$ triples. *ID* is the sensor identifier, denoting indirectly the type of data, while *value* is the data point.

---

[7] http://kaspar.informatik.uni-freiburg.de/∼osm/

[8] Developed using Android SDK Tools, Revision 21.1, corresponding to Android Platform version 4.2.2 (API level 17)

– To evaluate the variability of the information collected in the monitored area, mean, variance and deviation of data in the current time window are calculated.
– Incremental ratios of the above indexes are calculated on the same time windows, in order to detect significant changes in the collected data, which can be traced back to relevant events.
– A (binary or multiple) classifier is exploited to detect relevant events when given conditions occur, for every data stream. The identification is performed by taking into account threshold values for statistical indexes.
– Finally a logic-based annotation referred to knowledge modeled in a proper ontology (formalizing a conceptualization of the sensing domain) is constructed as the logical conjunction of all expressions derived from sensor data.

IV. CASE STUDY: COLLABORATIVE SENSING FOR ROAD STATE MONITORING

A toy example referred to cooperative environmental risk monitoring was developed to highlight capabilities of the proposed framework. The scenario is configured as a Hybrid Sensor and Vehicular Network (HSVN), with sensors distributed along roads to monitor and gather information about the environmental conditions of a given area. By means of Vehicle-to-Infrastructure (V2I) wireless communication technologies, each vehicle receives safety warnings and traffic information from Road-Side Units (RSUs). Each RSU acts as a CoAP gateway and periodically queries the CoAP sinks in its range, which are responsible for selecting and returning the most suitable sensors set through semantic-based resource discovery. The RSU can then start obtaining raw data from sensors and exploits data mining, as described in Section III-B. Event annotations are then exposed for external applications. Referring to parameters in Table II, a total of three RSUs, eight sinks and fourteen sensors are in the network, with maximum distance of 100 m between a sensor and its sink and 1000 m between a sink and its RSU. As reported in [17], the SSN-XG ontology [4] was extended to enable the specification of both observed parameters (*e.g.*, temperature, humidity, atmospheric pressure, wind speed) and sensor measurement capabilities (*e.g.*, accuracy, precision, resolution, frequency) as conjunctive concept expressions, following the *Stimulus-Sensor-Observation* design pattern [4].

Let us consider *a car travelling in the morning on SS16, a highway near to Bari, in Italy. The air humidity level is high and the pressure values are low. Furthermore, the road has low-density traffic with less than 100 vehicles flowing per hour. Possible risks are due to crossroads*. The scenario was randomly generated by the JOSM plugin presented in Section III-A: as shown in Figure 3 three RSUs, eight sinks and fourteen sensors are in the configured network scenario. The *RSU1* gateway composes a discovery request R, using concepts defined in the domain ontology, as reported in Figure 4 using OWL 2 *Manchester Syntax*. For the sake of readability, concept expressions for both request and sensors are summarized in textual form. The CoAP `GET` request also includes: (i) the RSU reference location P, defined through attributes `lt` and `lg`; (ii) maximum distance `md`; (iii) minimum covering threshold `sr` for resource retrieval. *RSU1* looks for devices located near to SS16 with a maximum distance of 2500 m from P and a coverage threshold value of 95%. After a distance-based pre-filtering, RSU1 solves the CCoP [3].

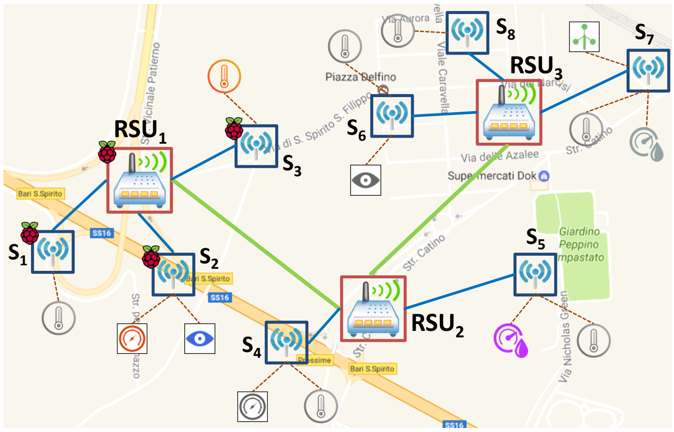


Fig. 3. Case study for HSVN configuration

Figure 4 shows semantic descriptions for some of the sensors inside the measurement area in Figure 3 and connected to gateway node RSU1. Based on the required measurement features, connected sinks retrieve a covering set $S_c(RSU1)$ composed of *TSic306Sensor*, *BMP085Sensor* and *FS11Sensor*, which does not fully cover the request. For this reason the uncovered part U_{RSU1} is returned, corresponding to the 37% of R. In particular, no anemometer or humidity sensor has been retrieved, while *TSic306Sensor* and *BMP085Sensor* do not completely satisfy the required features in terms of temperature and pressure measurement capabilities. Accordingly, RSU1 sends a CoAP semantic request to each reachable gateway (in the reference scenario, RSU2), forwarding U_{RSU1} to possibly discover remaining sensors. Based on its configuration, $S_c(RSU2)$ is composed by *HyteModSensor*, while U_{RSU2} is 14%. Likewise, *RSU2* sends U_{RSU2} to RSU3, obtaining *BitLineBLVSensor*. Finally, RSU2 returns the discovered sensor set to RSU1, also specifying the final uncovered part U_{RSU3}, corresponding to 5% of the original R.

Now RSU1 acquires data from the retrieved sensors for weather event detection. After a period of observation, the mining process described in Section III-B identifies *Fog* and *Rain* events (computation details not reported due to lack of space). Each detected event is annotated w.r.t. the reference ontology as subclass of the *Weather* concept and in terms of safety requirements a car must implement to limit risks (Figure 5). RSU1 waits for vehicles equipped with a wireless interface entering its radio range. Let us suppose the vehicles described in Figure 6 drive nearby RSU1 and are equipped with the prototypical system for real-time monitoring and driving assistance described in [20]. This means each vehicle is able to interpret data extracted from On-Board Diagnostics (OBD-II) car interface and smartphone micro-devices integrating local environmental information and potential risk factors

R (request) ≡ Sensor and (hasTemperature only (MediumRes. and HighRange. and HighLaten.)) and (hasVisibility only (MediumAcc. and LowFreq.)) and (hasOperatingRange only LowMedAltit.) and (hasPressure only (LowAcc. and MediumRes.)) and (hasWindSpeed only (LowRes. and MediumAcc. and MediumPrec.)) and (hasHumidity only (HighAcc. and HighRes. and MediumFreq.))

TSic306Sensor (S_1) ≡ TemperatureSensor and (hasTemperature only (LowAcc. and HighRange and MediumRes. and LowLaten.))) and (hasOperatingRange only LowMedAltit.)

LM70Sensor (S_3) ≡ TemperatureSensor and (hasTemperature only (LowAcc. and HighRange and LowRes.)) and (hasOperatingRange only LowAltit.)

BMP085Sensor (S_2) ≡ Barometer and (hasPressure only (LowAcc. and LowRes. and HighRange and LowPrec.))

FS11Sensor (S_2) ≡ VisibilitySensor and (hasVisibility only (MediumAcc. and MediumRange and LowFreq. and HighLaten.))

HyteModSensor (S_5) ≡ HumiditySensor and (hasHumidity only (HighAcc. and HighRes. and HighRange and MediumFreq.))

BitLineBLVSensor (S_7) ≡ AnenometerSensor and (hasWindSpeed only (MediumAcc. and LowRes. and MiddleRange and LowPrec.))

Fig. 4. Request and sensors descriptions

Fog ≡ Weather and (hasSpeed only ModerateSpeed) and (hasLamp only FogLamp) and (hasSafetyDevice only ABS)

Rain ≡ Weather and (hasSpeed only ModerateSpeed) and (hasSafetyDevice only (ABS and ESP)) and (hasPneumatic only RibbedTire)

Fig. 5. Weather events descriptions

in a proper annotation. Each RSU can use the information received from cars to further enrich event descriptions, *e.g.*, with road surface conditions and traffic level.

As soon as a vehicle comes into the gateway radio coverage, RSU1 will exploit the Concept Abduction inference service [3] to infer the vehicle risk level (*RL*) w.r.t. the detected events: very low ($0 \leq RL \leq 1$), low ($RL = 2$), medium ($RL = 3$), high ($4 \leq RL \leq 5$), very high ($RL = 6$), ultra high ($RL \geq 7$). As evidenced in Table III, the safest vehicle is Mercedes GLC, because it is equipped with snow tires (also useful in case of rain), fog lamps, ABS and ESP. The Opel Astra has higher risk level in case of rain due to its high speed and the absence of ribbed tire that contributes to increase the risk level in a significant way, despite the activated ABS and fog lamps. Absolutely inadequate for the detected weather conditions are the safety equipment and the high speed of the Toyota Aygo, indeed it has the highest risk levels.

Mercedes_GLC ≡ Vehicle and SUV and (hasSpeed only ModerateSpeed) and (hasLamp only (XenonLamp and FogLamp)) and (hasSafetyDevice only (ABS and ESP)) and (hasPneumatic only SnowTire)
Mercedes_GLC_Sensed ≡ (trafficLevel only Low) and (roadSurface only Irregular)

Opel_Astra ≡ Vehicle and Sedan and (hasSpeed only HighSpeed) and (hasLamp only FogLamp) and (hasSafetyDevice only ABS) and (hasPneumatic only SlickTire)
Opel_Astra_Sensed ≡ (trafficLevel only Low) and (roadSurface only SlightlyIrregular)

Toyota_Aygo ≡ Vehicle and EconomyCar and (hasSpeed only HighSpeed) and (hasLamp only HeadLamp) and (hasPneumatic only SlickTire)
Toyota_Aygo_Sensed ≡ (trafficLevel only Low) and (roadSurface only Irregular)

Fig. 6. Vehicles semantic annotations

TABLE III
VEHICLE RISK LEVELS FOR DETECTED EVENTS

	Mercedes_GLC	**Opel_Astra**	**Toyota_Aygo**
RL_{Fog}	very low (1)	low (2)	very high (6)
RL_{Rain}	very low (1)	high (4)	very high (7)

TABLE IV
TIME PERFORMANCE EVALUATION

	RSU1	**RSU2**	**RSU3**
T_i **(ms)**	1232	7	14
T_c **(ms)**	355	25	22
T_f **(ms)**	93	29	–

V. EXPERIMENTAL RESULTS

Effectiveness of the proposed appraoch was evaluated measuring data transfers and the time RSU1 required to retrieve sensors suitable for road environment monitoring. Semantic-enhanced CoAP servers were deployed on Raspberry Pi[9] single-boards computers for RSU1 and its three sinks. The JOSM plugin running on a workstation[10] simulated RSU2 and RSU3 gateways, as well as the sinks and sensors they manage. Reasoning services were provided by the embedded *Mini-ME* mobile matchmaker [3]. Time and data exchange results in the simulated scenario were measured. With reference to times, five runs of each test were executed and the mean of the last four runs was taken. Results are reported in Table IV: waiting time (T_i); time for computing Concept Covering (T_c); the time (T_f) after forwarding the uncovered part of the request. Results show a performance gap between the real node (RSU1) and simulated ones (RSU2, RSU3), due to processing and memory constraints on the Raspberry Pi. However, RSU1 turnaround time ($\simeq$ 1.7 s) appears satisfactory. The longest step is T_i, which includes data structures creation by the embedded reasoning engine. As found in [3], T_c on the Raspberry Pi node is roughly an order of magnitude higher than RSU2 and RSU3. Table V shows data exchange outcomes: total bytes transmitted and received (B_{itx}, B_{irx}) to obtain sensors descriptions and bytes exchanged (B_{ftx}, B_{frx}) with the other RSUs forwarding the uncovered part of the request. B_{itx} is always significantly shorter, because it contains only standard CoAP queries. Conversely, sink replies RSU-to-RSU packets include compressed semantic descriptions [17].

Comparing standard and semantic CoAP request/response pairs is useful to assess the proposed CoAP enhancement. A test was executed with the real nodes only, with a semantic request concerning operating range and temperature measurement capabilities as in Figure 4; for standard CoAP, only the query parameter `rt="Temperature"` was used to search for temperature sensors and the reply just contains a resource list. Figure 7 reports turnaround time, transmitted

---

[9] Raspberry Pi Model B (https://www.raspberrypi.org/products/model-b/), equipped with Wheezy Raspbian (http://www.raspbian.org) operating system

[10] HP Z820 with Xeon E5-2643 quad-core CPU at 3.3 GHz, 16 GB RAM, 1 TB hard disk

TABLE V
NETWORK PERFORMANCE EVALUATION

	RSU1	**RSU2**	**RSU3**
B_{itx} **(byte)**	36	24	36
B_{irx} **(byte)**	2004	2004	2992
B_{ftx} **(byte)**	1215	1154	–
B_{frx} **(byte)**	1720	1195	–

Fig. 7. Basic CoAP vs semantic CoAP

(TX) and received (RX) data results. Although heavier in terms of processing time and packet size, the semantic-enhanced CoAP protocol allows to specify a more precise query and obtain an accurate and smaller response, listing resources in relevance order with related metadata.

VI. CONCLUSION AND FUTURE WORK

The paper proposed a novel Semantic Sensor Network approach, featuring cooperative sensing and data stream mining. It included backward-compatible CoAP enhancements for semantic-based resource description, management and discovery. A HSVN case study and experiments on a prototypical implementation validated benefits and feasibility of the proposal. As expected, processing times and network load were slightly higher w.r.t. standard CoAP, but the improved quality of discovery can justify the approach in advanced, large-scale scenarios like distributed environmental monitoring.

Future work includes: (i) improving the semantic-based data mining through the integration of machine learning techniques; (ii) adopting a more expressive logic language to allow more articulated resource and request descriptions (iii) larger testbed implementation and experimental evaluation.

ACKNOWLEDGEMENTS

The authors acknowledge partial support of Apulia region cluster project PERSON.

REFERENCES

[1] M. Ruta, F. Scioscia, and E. Di Sciascio, “Enabling the Semantic Web of Things: framework and architecture,” in *Sixth IEEE International Conference on Semantic Computing (ICSC 2012)*, 2012, pp. 345–347.

[2] C. Bormann, A. P. Castellani, and Z. Shelby, “Coap: An application protocol for billions of tiny internet nodes,” *Internet Computing, IEEE*, vol. 16, no. 2, pp. 62–67, 2012.

[3] M. Ruta, F. Scioscia, E. Di Sciascio, F. Gramegna, and G. Loseto, “Mini-ME: the Mini Matchmaking Engine,” in *OWL Reasoner Evaluation Workshop (ORE 2012)*, ser. CEUR Workshop Proceedings, vol. 858. CEUR-WS, 2012, pp. 52–63.

[4] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog *et al.*, “The SSN ontology of the W3C semantic sensor network incubator group,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, pp. 25–32, Dec. 2012.

[5] W. Colitti, K. Steenhaut, N. De Caro, B. Buta, and V. Dobrota, “REST Enabled Wireless Sensor Networks for Seamless Integration with Web Applications,” in *8th IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS 2011)*, 2011, pp. 867–872.

[6] A. Dimitrios, G. Vasileios, G. Dimitrios, and C. Ioannis, “Employing Internet of Things technologies for building automation,” in *17th IEEE Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, 2012, pp. 1–8.

[7] H. A. Khattak, M. Ruta, and E. Di Sciascio, “CoAP-based healthcare sensor networks: A survey,” in *Applied Sciences and Technology (IBCAST), 2014 11th International Bhurban Conference on*. IEEE, 2014, pp. 499–503.

[8] A. Llaves, O. Corcho, P. Taylor, and K. Taylor, “Enabling RDF Stream Processing for Sensor Data Management in the Environmental Domain,” *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 12, no. 4, pp. 1–21, 2016.

[9] D. J. Russomanno, C. R. Kothari, and O. A. Thomas, “Building a Sensor Ontology: A Practical Approach Leveraging ISO and OGC Models,” in *The 2005 International Conference on Artificial Intelligence*, 2005, pp. 637–643.

[10] D. Pfisterer, K. Romer, D. Bimschas, O. Kleine, R. Mietz, C. Truong, H. Hasemann, A. Kroller, M. Pagel, M. Hauswirth *et al.*, “SPITFIRE: Toward a Semantic Web of Things,” *Communications Magazine, IEEE*, vol. 49, no. 11, pp. 40–48, 2011.

[11] P. Desai, A. Sheth, and P. Anantharam, “Semantic Gateway as a Service Architecture for IoT Interoperability,” in *2015 IEEE International Conference on Mobile Services*, June 2015, pp. 313–319.

[12] P. Barnaghi, M. Presser, and K. Moessner, “Publishing linked sensor data,” in *3rd International Workshop on Semantic Sensor Networks (SSN), Organised in conjunction with the International Semantic Web Conference*, ser. CEUR Workshop Proceedings, vol. 668, 2010.

[13] D. Le-Phuoc, H. Q. Nguyen-Mau, J. X. Parreira, and M. Hauswirth, “A middleware framework for scalable management of linked streams,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 16, pp. 42–51, Nov. 2012.

[14] K.-N. Tran, M. Compton, and R. G. Jemma Wu, “Semantic Sensor Composition,” in *3rd International Workshop on Semantic Sensor Networks. Proceedings of the 9th International Semantic Web Conference (ISWC 2010)*, ser. CEUR Workshop Proceedings, D. R. D. Taylor K., Ayyagari A., Ed., vol. 668. CEUR-WS, nov 2010, pp. 33–48.

[15] C. Perera, A. Zaslavsky, C. Liu, M. Compton, P. Christen, and D. Georgakopoulos, “Sensor Search Techniques for Sensing as a Service Architecture for the Internet of Things,” *Sensors Journal, IEEE*, vol. 14, no. 2, pp. 406–420, 2014.

[16] K. Taylor and L. Leidinger, “Ontology-driven complex event processing in heterogeneous sensor networks,” *The Semanic Web: Research and Applications*, pp. 285–299, 2011.

[17] M. Ruta, F. Scioscia, A. Pinto, E. Di Sciascio, F. Gramegna, S. Ieva, and G. Loseto, “Resource annotation, dissemination and discovery in the Semantic Web of Things: a CoAP-based framework,” in *Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE Int. Conf. on and IEEE Cyber, Physical and Social Computing*. IEEE, 2013, pp. 527–534.

[18] M. Kovatsch, S. Mayer, and B. Ostermaier, “Moving Application Logic from the Firmware to the Cloud: Towards the Thin Server Architecture for the Internet of Things,” in *Proceedings of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS 2012)*, Jul. 2012.

[19] M. Gobelbecker and C. Dornhege, “Realistic Cities in Simulated Environments - An Open Street Map to Robocup Rescue Converter,” in *4th International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster (SRMED 2009)*, 2009.

[20] M. Ruta, F. Scioscia, F. Gramegna, and E. Di Sciascio, “A Mobile Knowledge-based System for On-Board Diagnostics and Car Driving Assistance,” in *4th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, 2010, pp. 91–96.