

Semantic-enabled Resource Discovery, Composition and Substitution in 802.11 Pervasive Environments

M. Ruta, G. Zacheo, A. Grieco, T. Di Noia, G. Boggia, E. Tinelli, P. Camarda, E. Di Sciascio
Politecnico di Bari, Dipartimento di Elettrotecnica ed Elettronica
I-70125 Bari, Italy
{m.ruta, g.zacheo, a.grieco, t.dinoia, g.boggia, e.tinelli, camarda, disciascio}@poliba.it

Abstract

We present a general framework for resource discovery, composition and substitution in wireless networks, exploiting knowledge representation techniques: discovery information are reused at network layer to build a fully unified discovery and routing framework. Performances of the proposed framework have been evaluated using ns-2 simulator with reference to a disaster recovery scenario.

1 Introduction

Mobile Ad-hoc NETWORKS (MANETs) are made by smart nodes, equipped with Wireless Network Interface Cards, able to automatically build up a multi-hop communication infrastructure when placed in the same area [8]. Basically, MANETs are employed in scenarios where it is not possible or convenient to deploy a wired network infrastructure, e.g., military or disaster recovery operations. They have to provide flexibility, fault tolerance, and ability of self-configuration: often cross-layer protocols help in implementing these features [7]. In this paper we investigate the issue of integrating routing and discovery protocols in MANETs, using a novel semantic-based approach. Given a user request and a set of available services/resources both described using a subset of Ontology Web Language OWL-DL¹, the framework allows us to carry out discovery and composition of mobile resource components covering as much as possible the needs of the requester. To increase the fault tolerance of the system, our composition protocol integrates the substitutability among resource components. A data dissemination mechanism has been also devised. It allows to advertise in broadcast only resource identifiers throughout the network permitting to locate and download in unicast the semantic annotations when needed. Using

ns-2² simulations, we have evaluated performances of the framework w.r.t. a disaster recovery scenario. Particularly, the impact of the available services and active clients has been investigated in many mobility conditions and for several covering threshold levels. Results have clearly pointed out the effectiveness of the approach.

The remaining of the paper is organized as follows: in next Section we describe both data dissemination protocol and proposed cross-layer routing/discovery framework. Section 3 presents composition and substitution features our approach provides. An illustrative example is presented in Section 4 followed by a thorough performance evaluation. Conclusions terminate the paper.

2 Data Dissemination Framework

The framework we propose exploits a controlled dissemination of advertisements containing resource locations followed by an “on demand” download of resource descriptions. Only information strictly required for the unambiguous identification of a resource is diffused within the network whereas the semantic annotation of the resource itself will be sent in unicast only to nodes which have to perform a composition. We hypothesize each resource in the MANET is unambiguously identified by means of the triple [*SOURCE ADDRESS*, *OUUID*, *RESOURCE ID*], where the first value is the IP address of the node hosting a resource, the second one stands for *Ontology Universally Unique Identifier* and marks the specific reference ontology the resource annotation is associated with, the last one is a value to distinguish different resources coming from the same host node and referred to the same ontology. As in [10], we postulate the existence of a unique OUUID for each ontology, thus allowing an unambiguous identification in the whole Semantic based Web. Every 2000 ms, a mobile node hosting a resource broadcasts an advertisement and nearby nodes forward the packet; as a consequence, a resource

¹<http://www.w3.org/TR/owl-features/>

²ns-2, the network simulator– <http://www.isi.edu/nsnam>

provider listen to the echo of the advertisement packet it originally transmitted. Thus, it can obtain a confirmation of the presence of other nodes in its neighborhood. If no echoes are received within 500 ms, it will retransmit the advertisement, assuming that a collision or a transmission error has occurred. After 5 retries, it can be assumed there are no neighbors, so the transmission of the advertisement can be scheduled for a longer timeout to reduce power consumption. The proposed approach integrates routing, and discovery levels, so that routing sequence numbers are exploited in the discovery phase in order to verify if the information about a resource is up to date. Moreover, the routing protocol updates its tables using discovery PDUs, even if a *Route Discovery* session has not been started. When a node receives an advertisement, it extracts the routing information, as explained later on. Then, if the resource was previously unknown, a new entry may be created in the host cache; otherwise the node, before updating stored data, verifies if the information received is more recent or has ran across a shorter path than the existing one. If the cache is updated and the maximum advertisement travel (expressed in hops number) has not been reached it is forwarded. Otherwise, the whole packet will be silently discarded. In order to reduce the collision probability (MAC 802.11 protocol does not provide any acknowledgment for broadcasting), each host waits a random time t before transmitting, with $t \in [0, 40]ms$.

A node starting a service composition, firstly checks for possible compatible entries within its own cache and, in case, it retrieves in unicast the corresponding semantic annotations by means of specific demand PDUs sent to providers. A client requires all missing resource descriptions and then waits for replies up to $2000 ms \cdot max_distance$ ms (where *max_distance* is the maximum hop number between requester and each provider). When this time has expired or all requested PDUs have been received, the requester starts the composition. The composition result (covering level) ranges from 0 to 100% in case of fully covered request. The requester imposes a minimum covering threshold she wants and, if the the cmposition outcome is under the given threshold, a node should require further descriptions in order to attempt a new orchestration. Thus, it will transmit a solicit packet to nearby nodes with a mechanism similar to the advertising one. A node generating a solicit waits for an acknowledgment from each neighbor for 50 ms. Before starting the composition or requesting new descriptions, each node wait for the expected cache content PDUs for a time t_w defined as:

$$t_w = 2000ms + 50ms \cdot (current_hops - 1) \quad (1)$$

where *current_hops* is the hops number the solicit has still to traverse to reach the maximum discovery diameter. This procedure is repeated by all forwarding nodes along the solicit path up to 4 hops. Each node

manages a cache table storing resource information and when receiving a solicit PDU, it replies in unicast with a cache content PDU toward the node the solicit came from.

Interactions between Service Discovery and Routing protocols.

AODV [6] is a widespread Distance Vector routing protocol which has been chosen as reference routing protocol. When a route toward a destination is needed and an active one does not exist yet (or it has expired), AODV starts a path discovery procedure broadcasting *Route Request* packets [6]. The destination node replies with unicast frames processed hop by hop in order to create an active bidirectional route. Moreover, whenever a node detects a broken link, it can start either a *Route Repair* process or a new *Route Discovery*. Nevertheless, MAC protocols for MANETs do not provide any handshake mechanism for broadcasting, then the transmitting node ignores if *Route Request* messages have been actually delivered during the path discovery phase. Hence *Route Request* packets are periodically sent until a timer expires or a *Route Reply* packet is received. Also resource discovery protocols can use multicast/broadcast messages during the advertisement/discovery phase [4] in order to propagate information about all the available resources. In the proposed approach, routing information are piggybacked in the advertisement, solicit, demand, and cache content PDUs. In this way, paths toward providers are pro-actively set, thus minimizing latencies due to the route creation phase. Each node receiving an advertisement is primarily able to build an active route toward the provider at one hop distance. Furthermore, also a node receiving a forwarded advertisement can build a route toward a provider (by exploiting the provider sequence number and the value of *traveled_hops*). When the advertisement phase has been accomplished, each node which has received an advertisement maintains an active route toward every resource provider within the advertisement travel range. In the same way, if a node receiving a solicit manages one or more cache entries matching contained parameters, it replies with a cache content PDU to the node the solicit comes from. When a node receives the packet, it updates its routing table building a route toward the sender. Moreover, since cache content PDUs are not simply forwarded but processed at every hop by both routing and discovery protocols, every involved node will build a route toward each known provider.

3 Semantic-based Resource Composition and Substitution

We assume the reader be familiar with basics of Semantic Web and of Description Logics (DLs) [1], the formal language we adopt here. We define both the request D and the description of each available resource within

the network, as DL concepts both referred to an ontology \mathcal{T} shared among some users. We extend the service composition model in [3] to deal with a pervasive scenario. For the sake of clarity, we only recall main terms and definitions. In particular we define as **Mobile Resource** a triple $\langle MR_D, P, E \rangle$ where MR_D is the description of the provided resource, P its preconditions and E the effects. Furthermore, indicating with AI_i the available information for the i -th mobile resource mr_i and with E_j the effects produced by mr_j , with $j < i$, the following relation ensues: $AI_i = P_0 \sqcap E_1 \sqcap E_2 \sqcap \dots \sqcap E_{i-1}$. Based on the definition of mobile resource flow w.r.t. some initial preconditions P_0 –from now on $\mathcal{MSF}(P_0)$ [3]– here we define a **composite mobile resource** w.r.t. a request D . In particular, a composite mobile resource for $\langle D, P_0 \rangle$ w.r.t. the set of discovered resources \mathcal{R} , from now on $\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)$, is a mobile resource flow such that for each mr_j in the execution flow: $D_{\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)} = \{MR_D(j) | mr_j \in \mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)\}$ covers D . An **executable mobile resource** mr^{ex} for $\mathcal{MSF}(P_0)$ is a mobile resource which can be invoked after the execution of $\mathcal{MSF}(P_0)$, i.e., its preconditions are satisfied after the execution of $\mathcal{MSF}(P_0)$, and such that its effects are not already provided by $\mathcal{MSF}(P_0)$. The composition algorithm (see later on) is the core of the system. In input we submit preconditions that must be satisfied as well as the maximum discovery distance and the minimum covering threshold [9]. The first composition attempt is performed using resources in the requester’s cache. The algorithm outputs the \mathcal{CMR} as well as the uncovered part of the request $D_{uncovered}$. If the covering level is under the threshold, the uncovered demand as well as the temporary resulting \mathcal{CMR} are stored and the requester broadcasts a solicit packet in an expanding ring fashion for requiring new resource descriptions.

Algorithm *resourceComposer*($\mathcal{R}, \langle D, P_0 \rangle, \mathcal{T}$)

input a set of resources $\mathcal{R} = \{mr_i = \langle MR_D(i), P_i, E_i \rangle\}$, a request $\langle D, P_0 \rangle$ where D and $MR_D(i)$ are satisfiable in \mathcal{T}
output $\langle \mathcal{CMR}, H \rangle$

```

1 begin algorithm
2  $\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle) = \emptyset;$ 
3  $D_{uncovered} = D;$ 
4  $H_{min} = D;$ 
5 do
6 compute  $\mathcal{E}\mathcal{X}_{\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)};$ 
7  $MR_{Dmin} = \top;$ 
8 for each  $mr_i \in \mathcal{E}\mathcal{X}_{\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)}$ 
9 if  $\mathcal{D}_{\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)} \cup \{MR_D(i)\}$  covers  $D_{uncovered}$  then
10  $H = solveCAP(\langle \mathcal{L}, MR_D(i), D_{uncovered}, \mathcal{T} \rangle);$ 
11 if  $H \prec H_{min}$  then
12  $MR_{Dmin} = MR_D(i);$ 
13  $H_{min} = H;$ 
14 end if
15 end if
16 if  $MR_{Dmin} \neq \top$  then
17  $\mathcal{R} = \mathcal{R} \setminus mr_i;$ 
18  $\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle) = (\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle), mr_i);$ 
19  $D_{uncovered} = H_{min};$ 
20 end if
21 end for each
22 while  $(MR_{Dmin} \neq \top);$ 

```

```

23 return  $\langle \mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle), D_{uncovered} \rangle;$ 
24 end algorithm

```

Note that, in a pervasive environment, often it could be necessary to automatically and dynamically substitute resource components no more available with new ones. In order to implement this feature, here we define a *Similarity Group* (\mathcal{SG}) (a.k.a substitutability class) as a collection of components which can be substituted with each other according to a set of substitution rules [2]. An \mathcal{SG} is created w.r.t. each resource in the $\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)$. Information about resource interfaces (i.e., required preconditions and provided effects) are needed prior to admit it in a substitutability class [5]. This is mandatory to evaluate its correct insertion in the $\mathcal{MSF}(P_0)$. In order to decide if a generic resource can belong to an \mathcal{SG} , two conditions about preconditions and effects have to be verified. Let us suppose (mr_1, \dots, mr_N) are services in a $\mathcal{CMR}(\langle D, P_0, \mathcal{R} \rangle)$ and we have to create the mr_i similarity group $\mathcal{SG}(i)$. We say $mr_j^{sub} (j = 1 \dots L) \in \mathcal{SG}(i)$ iff the following conditions hold: **(1):** mr_j^{sub} is an *executable mobile resource* for $(mr_1, mr_2, \dots, mr_{i-1})$; **(2):** for $h = i + 1, \dots, N$, mr_h is an *executable mobile service* for $(mr_1, mr_2, \dots, mr_{h-1})$.

In a more compact way, to verify if a substitute resource mr_j^{sub} is suitable, we take into account the already available AI_i (if mr_i is the resource to substitute) and we recalculate next AI_k (with $k = i + 1 \dots N$) checking the satisfiability of the respective mr_k . In other words, if AI_i is the available information for mr_i and if we want to substitute the same mr_i , the new $AI'_{i+1} = AI_i \sqcap E_j^{sub}$ has to be determined. Hence, it will be used for checking the satisfiability of the next mr_{i+1} and so on. If one of these checks fails, we can conclude that the mr_j^{sub} is unsuitable. The similarity group of the resource mr_i is more and more enriched while the discovery progresses. Hence, if $\mathcal{SG}^k(i)$ is the similarity group of the service m_i at hop k , $\mathcal{SG}(i) = \mathcal{SG}^1(i) \cup \mathcal{SG}^2(i) \cup \dots \cup \mathcal{SG}^{MAX_{hop}}(i)$.

4 Case study

“An accident has occurred in a chemical plant during the night. An explosion has set the building on fire, and workers are trapped in. A robot team has been sent to search and rescue survivors. Besides fire, environmental hazards are not precisely known a priori but may include toxic gases and the presence of debris, liquid pools and narrow tunnels. The robot team is coordinated by a mobile headquarter deployed at short distance from the disaster area. Communication is provided by an ad-hoc IEEE 802.11 network. Energy has been properly supplied to robot units by means of batteries and/or fuel. The headquarter is the sink of collected data by the sensor array of robot units deployed in the field, providing information processing through dedicated computational resources such as

Physical_quantity ⊆ T	Mass ⊆ Physical_quantity
Temperature ⊆ Physical_quantity	Vibration ⊆ Physical_quantity
Concentration ⊆ Physical_quantity	Relative_humidity ⊆ Concentration
Angle ⊆ Physical_quantity	Longitude ⊆ Angle
Latitude ⊆ Angle	Length ⊆ Physical_quantity
Altitude ⊆ Physical_quantity	Distance ⊆ Length
Speed ⊆ Physical_quantity	Pressure ⊆ Physical_quantity
Connectivity ⊆ T	WLAN ⊆ Connectivity
WPAN ⊆ Connectivity	WiFi ⊆ WLAN
Bluetooth ⊆ WPAN	ZigBee ⊆ WPAN
Sensor ⊆ Vmeasures.Physical_quantity	Geographical ⊆ Sensor
Weather ⊆ Sensor	Scales ⊆ Sensor ∩ Vmeasures.Mass
Acoustic ⊆ Sensor ∩ Vmeasures.Vibration	Imaging ⊆ Sensor
Chemical ⊆ Sensor ∩ Vmeasures.Concentration	Rangefinder ⊆ Sensor ∩ Vmeasures.Distance
Air_analysis ⊆ Chemical	CO2air_analysis ⊆ Toxic_gas_analysis
SO2air_analysis ⊆ Toxic_gas_analysis	CO_gas_analysis ⊆ Toxic_gas_analysis
Water_analysis ⊆ Chemical	Toxic_metal_water_analysis ⊆ Water_analysis
Radar ⊆ Rangefinder	Sonar ⊆ Rangefinder
Laser_rangefinder ⊆ Rangefinder	Seismometer ⊆ Acoustic
Microphone ⊆ Acoustic	Videocamera ⊆ Imaging
Nocturnal_videocamera ≡ Videocamera ∩ Nighth_vision	Telephoto_lens ⊆ Imaging
Camera ⊆ Imaging	Color_camera ⊆ Camera
Black_white_camera ⊆ Camera	Anemometer ⊆ Weather ∩ Vmeasures.Speed
Barometer ⊆ Weather ∩ Vmeasures.Pressure	Hygrometer ⊆ Weather ∩ Vmeasures.Relative_humidity
Thermometer ⊆ Weather ∩ Vmeasures.Temperature	
Computational_resource ⊆ T	Processor ⊆ Computational_resource
DSP ⊆ Processor	CPU ⊆ Processor
Realtime_guarantee ⊆ Computational_resource	Soft_realtime ⊆ Realtime_guarantee
Hard_realtime ⊆ Realtime_guarantee	
Obstacle ⊆ T	Step ⊆ Obstacle
Slope ⊆ Obstacle	Debris ⊆ Obstacle
Pool ⊆ Obstacle	Hole ⊆ Obstacle
Tunnel ⊆ Obstacle	
Movement ⊆ Vovercomes_obstacle ∩ Vovercomes_obstacle.Obstacle	Water_movement ⊆ Movement ∩ Vovercomes_obstacle.Pool
Ground_movement ⊆ Movement ∩ Vovercomes_obstacle.(Tunnel ∩ Slope)	Wheels ⊆ Ground_movement
Caterpillar_track ⊆ Ground_movement	Amphibious_movement ≡ Ground_movement ∩ Water_movement
Action ⊆ T	Push ⊆ Action
Light ⊆ Action	Pierce ⊆ Action
Pull ⊆ Action	Extinguish_fire ⊆ Action
Process ⊆ Action	Lift ⊆ Action
Cut ⊆ Action	Plough_through_debris ⊆ Action
Detection ⊆ Action	Fire_detection ⊆ Detection
Presence_detection ⊆ Detection	Mine_detection ⊆ Detection
Actuator ⊆ Vperforms_action ∩ Vperforms_action.Action	Torch ⊆ Actuator
Signalling ⊆ Actuator	Dozer_blade ⊆ Actuator ∩ Vperforms_action.Push
Drill ⊆ Actuator ∩ Vperforms_action.Pierce	Hammer_drill ⊆ Drill
Forklift ⊆ Actuator ∩ Vperforms_action.Lift ∩ Vhas_capacity	Forklift_200kg ⊆ Forklift ∩ Vhas_capacity.(≤ 200 kg)
Pincers ⊆ Actuator	Fire_extinguisher ⊆ Actuator ∩ Vperforms_action.Extinguish_fire
Powder_fire_extinguisher ⊆ Fire_extinguisher	
Power_source ⊆ T	Internal_power_source ⊆ Power_source
Battery ⊆ Internal_power_source	Fuel ⊆ Internal_power_source
Outlet ⊆ Power_source	Outlet ⊆ -Internal_power_source
Supply ⊆ T	First_aid_kit ⊆ Supply

Figure 1. Disaster recovery ontology used in the case study

DSPs”. Each mobile unit has different capabilities, hence the overall search and rescue mission has to be divided into tasks which must be properly orchestrated. “Disaster management rules for the rescue mission include an air analysis, then exploration of the area (nocturnal visibility is required) by moving through debris. Fire sources should be detected and extinguished in order to proceed with exploration. Human presence must be detected and people have to be aided and extracted from the disaster area”. Fig. 1 reports a subset of ontology axioms modeling the reference domain. The above mission goal is the request for the service orchestration algorithm that has to find the most suitable composition of operational units. Similarly, facilities provided by the headquarter are modeled as the initial supplied preconditions. With respect to the ontology, they can be expressed as follows:

Demand: $D = Air_analysis \sqcap Night_vision \sqcap \exists performs_action \sqcap \forall performs_action.(Plough_through_debris \sqcap Detect_fire \sqcap Extinguish_fire \sqcap Detect_presence) \sqcap \exists overcomes_obstacle \sqcap \forall overcomes_obstacle.(Debris \sqcap Pool \sqcap Slope \sqcap Tunnel) \sqcap First_aid_kit$

Initial preconditions: $P_0 = WiFi \sqcap Fuel \sqcap Battery \sqcap DSP$

The composition algorithm can be applied. Preconditions and effects are described hereafter for each available robot unit in the mission area:

$ms_1 : Environmental_unit = \langle P_1, E_1 \rangle = \langle WiFi \sqcap Internal_power_source, GPS \sqcap Altimeter \sqcap Anemometer \sqcap Barometer \sqcap Hygrometer \sqcap Thermometer \sqcap CO_2air_analysis \sqcap SO_2air_analysis \sqcap CO_air_analysis \rangle$

$ms_2 : Meteorological_unit = \langle P_2, E_2 \rangle = \langle WiFi \sqcap Internal_power_source, GPS \sqcap Altimeter \sqcap Anemometer \sqcap Barometer \sqcap Hygrometer \sqcap Thermometer \rangle$

$ms_3 : Mine_unit = \langle P_3, E_3 \rangle = \langle Fuel, Wheels \sqcap Beacon \sqcap Loudspeaker \sqcap Metal_detector \sqcap \exists performs_action \sqcap \forall performs_action.(Mine_detection) \rangle$

$ms_4 : Fire_victim_detection_unit = \langle P_4, E_4 \rangle = \langle Battery \sqcap GPS \sqcap Videocamera \sqcap Microphone \sqcap Thermometer \sqcap CO_2Air_Analysis \sqcap DSP, \sqcap \exists performs_action \sqcap \forall performs_action.(Detect_presence \sqcap Detect_fire) \rangle$

$ms_5 : Rescue_unit = \langle P_5, E_5 \rangle = \langle \exists performs_action \sqcap \forall performs_action.(Detect_presence \sqcap Detect_fire), Pincers \sqcap$

$Powder_fire_extinguisher \sqcap First_aid_kit$
 $ms_6 : Scout_unit = \langle P_6, E_6 \rangle = \langle GPS \sqcap Altimeter \sqcap Anemometer \sqcap WiFi \sqcap Fuel, Amphibious.movement \sqcap Caterpillar_track \sqcap Hammer_drill \sqcap Forklift.200.kg \sqcap Nocturnal.videocamera \sqcap Laser.rangefinder \sqcap Microphone \rangle$

Let us suppose to assign a covering threshold of 90%. In what follows algorithm steps are reported.

$CMS = \emptyset$, $D_{uncovered} = D$, $Covering_Level = 0\%$

Step I: $\mathcal{E}\mathcal{X}(CMS) = \{ms_1, ms_2, ms_3\}$, $CMS = (ms_1)$

$D_{uncovered} = Night_vision \sqcap \exists performs_action \sqcap \forall performs_action.Plough_through_debris \sqcap Detect_fire \sqcap Extinguish_fire \sqcap Detect_presence \sqcap \exists overcomes_obstacle \sqcap \forall overcomes_obstacle.Debris \sqcap Pool \sqcap Slope \sqcap Tunnel \sqcap First_aid_kit$
 $Covering_Level = 9.1\%$

Notice that, among executable mobile resources, only ms_1 contributes to cover the demand since it provides resources for air analysis.

Step II: $\mathcal{E}\mathcal{X}(CMS) = \{ms_2, ms_3, ms_6\}$, $CMS = (ms_1, ms_6)$

$D_{uncovered} = \exists performs_action \sqcap \forall performs_action.Detect_fire \sqcap Extinguish_fire \sqcap Detect_presence \sqcap First_aid_kit$
 $Covering_Level = 63.6\%$

The mobile service unit ms_6 can now be triggered, since its required preconditions P_6 are satisfied (by E_1). It provides sensors and actuators that perform the required mission tasks for exploration of the disaster area. It can also be noticed that ms_2 would cause an unnecessary effect duplication with ms_1 .

Step III: $\mathcal{E}\mathcal{X}(CMS) = \{ms_2, ms_3, ms_4\}$, $CMS = (ms_1, ms_6, ms_4)$

$D_{uncovered} = \exists performs_action \sqcap \forall performs_action.Extinguish_fire \sqcap First_aid_kit$
 $Covering_Level = 81.8\%$

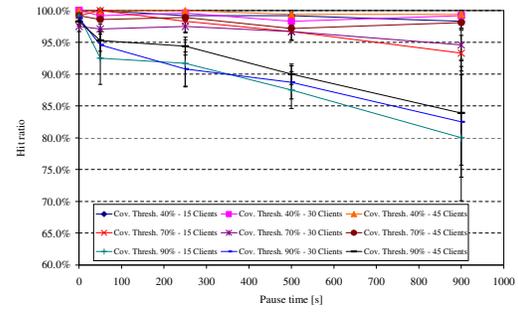
The mobile service unit ms_4 becomes executable and it is added to the composite flow because it provides resources for detection of fire sources and human presence.

Step IV: $\mathcal{E}\mathcal{X}(CMS) = \{ms_2, ms_3, ms_5\}$, $CMS = (ms_1, ms_6, ms_4, ms_5)$

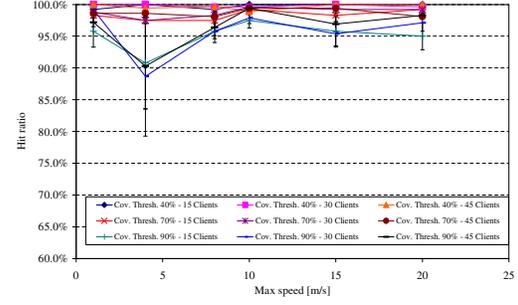
$D_{uncovered} = \top$
 $Covering_Level = 100\%$

Service unit ms_5 provides further required tools so it is selected. A full covering of the request has been reached and the composition now stops.

Performance Evaluation. The effectiveness of the proposed framework has been evaluated assuming a 50 robots rescue team moving in a 1000 square meters area, thus forming a MANET. At the application level we basically have a *hit* when the composition allows to surpass the covering threshold. In order to adapt this assertion to a parameter measurable at lower layers of the protocol stack, we simulated 250 service compositions for three different covering thresholds, *i.e.*, 40%, 70%, and 90%, exploiting 30 different individuals. Results showed that to reach a 90% covering threshold up to 7 component resources may be required, up to 6 components are needed to reach a 70% threshold whereas a 40% covering level needs up to 3 de-



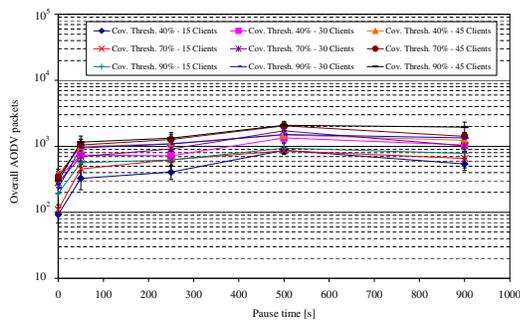
(a) W.r.t. pause time



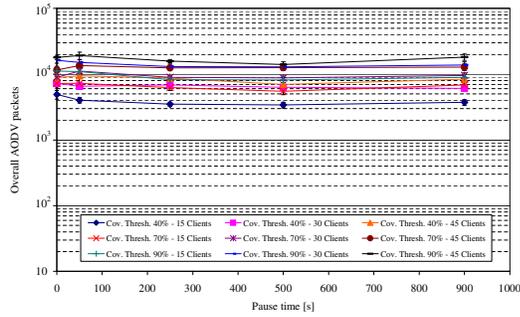
(b) W.r.t. maximum speed

Figure 2. Hit ratio

scriptions. Those values have been exploited to consider a hit in the following experiments. Node mobility is driven by the “random waypoint” model characterized by pause time and speed parameters. Simulations last 2700 seconds. The simulations are arranged in two sets: in the first one the speed time has been varied from 1 to 20 m/s while keeping the pause time fixed to 0.01 s; in the second set, the speed parameter has been set to 1 m/s while the pause time has been varied from 0 to 900 s. Simulation results point out the average number of partitions in each scenario is smaller than two. In both simulation sets, we have hypothesized 7, 9, and 11 nodes hosting resources (providers) and 15, 30 and 45 client nodes. For each combination of reference parameters we run 8 simulations exploiting different values for the seed of the ns-2 random number generator. Obtained results have been averaged. Fig. 2 reports the hit ratio as a function of the pause time and of the maximum speed, respectively. Satisfied requests are very high, with values ranging from 80% to 100%, also considering the challenging 90% covering threshold. If we consider the 70% covering threshold the hit ratio is always greater than 90%. As a general consideration, this result clearly indicates the relevance of the proposed approach. Further simulations proved the average time elapsed for obtaining a hit decreases by increasing the clients for a given covering threshold. This feature enforces the scalability properties. Finally, to prove the effectiveness of the cross-layer approach, in Figs. 3 and 4 the packets generated by the



(a) Interaction enabled



(b) Interaction disabled

Figure 3. AODV PDUs (w.r.t. pause time)

AODV protocol are reported. The overall routing overhead is reduced by at least 85%, reaching 99% in the scenarios with less mobility.

5 Conclusion

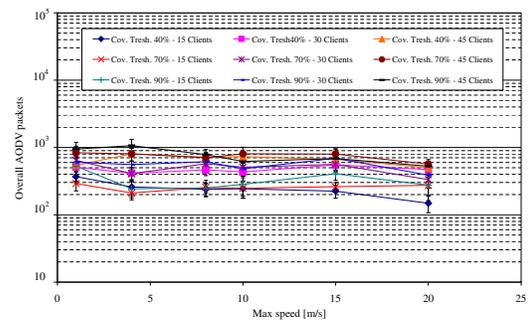
We have proposed an innovative framework to enable the resource discovery, composition and substitution in IEEE 802.11 ad-hoc networks exploiting semantic technologies. The approach allows us to reuse discovery information at network level to enable a cross-layer interaction. The approach has been tested within a ns-2 simulation environment with reference to a disaster recovery scenario.

Acknowledgments

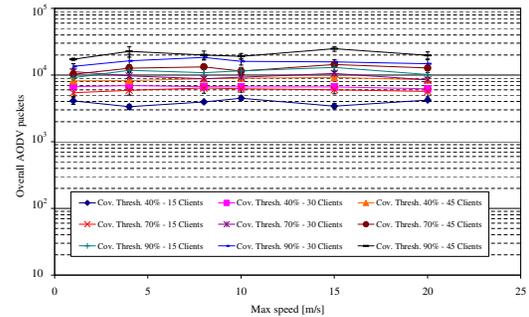
The authors acknowledge partial support of Apulia Region Strategic Projects PS_092 and PS_121.

References

- [1] F. Baader, D. Calvanese, D. Mc Guinness, D. Nardi, and P. Patel-Schneider. *The Description Logic Handbook*. Cambridge University Press, 2002.
- [2] L. Bordeaux, G. Salaün, D. Berardi, and M. Mecella. When are Two Web Services Compatible? In *TES, Lecture Notes in Computer Science*, pages 15–28. Springer, 2004.



(a) Interaction enabled



(b) Interaction disabled

Figure 4. AODV PDUs (w.r.t. maximum speed)

- [3] T. Di Noia, E. Di Sciascio, F. Donini, A. Ragone, and S. Colucci. Automated semantic web services orchestration via concept covering. In *WWW '05*, pages 1160–1161. ACM, 2005.
- [4] U. Kozat and L. Tassiulas. Service discovery in mobile ad hoc networks: an overall perspective on architectural choices and network layer support issues. *Ad Hoc Networks*, 2(1):23–44, 2004.
- [5] M. Mecella, B. Pernici, and P. Craca. Compatibility of e-services in a Cooperative Multi-platform Environment. In *TES, LNCS*, pages 44–57. Springer, 2001.
- [6] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561, 1994.
- [7] V. T. Raisinghani and S. Iyer. Cross-layer feedback architecture for mobile device protocol stacks. *IEEE Communications Magazine*, pages 85–92, January 2006.
- [8] R. Ramanathan and J. Redi. A brief overview of ad hoc networks: Challenges and directions. *IEEE Communications Magazine*, 40(5):20–22, 2002.
- [9] M. Ruta, T. Di Noia, E. Di Sciascio, and F. Donini. Semantic enabled Resource Discovery, Composition and Substitution in Pervasive Environments. In *13th IEEE MELECON 2006*, pages 754–760. IEEE, 2006.
- [10] M. Ruta, T. Di Noia, E. Di Sciascio, and F. Donini. Semantic Based Collaborative P2P in Ubiquitous Computing. *Web Intelligence and Agent Systems*, 5(4):375–391, 2007.