International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 9, No.5 (Sep-2020)

http://dx.doi.org/10.12785/ijcds/090514

Decentralized Service Discovery and Localization in Internet of Things Applications Based on Ant Colony Algorithm

Zahia BENSALAH AZIZOU[1], Abdelmalek BOUDRIES[2] and Mourad AMAD[1]

[1] LIMPAF Laboratory, Computer Sciences Dept, Faculty of Sciences and Applied Sciences, University of Bouira, Algeria
[2]LMA Laboratory, University of Bejaia, Algeria

Received 22 Nov. 2019, Revised 27 Feb. 2020, Accepted 25 Aug.. 2020, Published 1 Sep. 2020

Abstract: Internet of things is a technological revolution, it allows heterogeneous and smart objects to connect to the Internet, and then communicate and exchange the information between them. Emerging applications and services in the Internet of Things provided by different objects need a service discovery mechanism used by objects. However, the lookup of a desired service is quite challenging difficult, many centralized and distributed service discovery approaches have been investigated. Therefore, in this paper, we propose a new distributed discovering algorithm based on Ant Colony Algorithm (ACA) for IoT. ACA is a metaheuristic algorithm inspired by nature, to solve combinatorial optimization problem. To show the effective performance of ACA algorithm in accurate discovering of services, simulation results indicate that the proposed approach has a good performance with reduced number of hops used to discover services and can achieve a high discovery success rate. Our approach maintains their performance, even in a large scale network.

Keywords: Decentralized Service Discovery, Internet of Things, Ant Colony Lgorithm (ACA)

1. Introduction

Internet of things is a new paradigm, which allows heterogeneous objects to communicate using wireless technology such as WiFi, Bluetooth, IEEE 802.15.4, Z-wave, LTE-Advanced, RFID, Near Field Communication (NFC) and bandwidth ultra-wide (UWB). In Internet of things, each object offers functionality as services. With increasing of objects, there will be about 50 billion connected objects by 2020 according to Dave Evans [1]. However, the heterogeneity of objects, their capabilities (*limited energy, storage capabilities, computing power*) [2], mobility, properties, different communication technologies, and protocols brings many challenges and requirements to the service discovery.

The adoption of bio-inspired computing approach has attracted significant research [3][4][5]. This decentralized approach is very suitable for IoT, in particular to service discovery, with taking advantages of their inherent support for self-adaptively, self-organization and scalability. Heuristic search algorithms with self-organization take inspiration from some biological phenomena, offering scalability, fault tolerance and flexibility in unstructured organizations. Ant colony algorithm conforms to the characteristics of the service discovering.

In this paper, a decentralized service discovery model based on Ant Colony Algorithm (ACA) is presented. ACA is inspired by the behavior of ants to find food, they use the pheromone deposited on trails to find the short paths between their colony and food sources. This characteristic of ant colonies is exploited in this paper to discover and to localize a suitable set of services in an IoT network to satisfy a user's request and to find the solution with a controlled number of hops.

The remainder of the paper is structured as follows: Section 2 summarizes the background and related works. Section 3 elaborates more on the problem statement, while section 4 presents a brief introduction on ACA. Section 5 introduces the proposed approach with formulation for service discovery and localization based on ACA, followed by Section 6, which presents a discussion of the simulation results. Finally, we conclude and give some future works.

2. Background and Related Works

Internet of Things (IoT) is a network of devices, also denoted as smart objects, which have the ability to sense, compute and transfer data over the internet without any human intervention. Each IoT object can be used to

zahia.azizou@gmail.com, abdelmalekboudries@gmail.com, amad.mourad@gmail.com

provide or to request IoT services. According to the 2020 conceptual framework [6], IoT architecture needs four components as expressed by simple formula (IoT = Services + Data + Networks + Sensors). Implementation of IoT is based on an architecture consisting of several layers: from the field data acquisition layer, networking layer and application layer as shown in Fig. 1. To gather with application-layer protocols, suitable mechanisms for service and resource discovery should be defined.

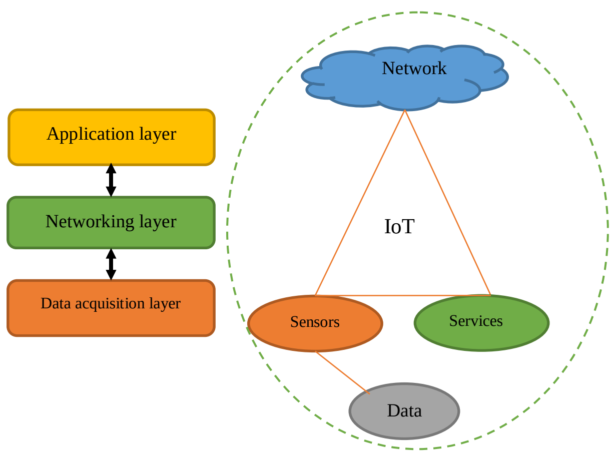


Figure 1. The IoT architecture

Service discovery is the action of finding and locating suitable services that can meet requesters' needs [7]. In IoT, there are generally two approaches for the service discovery protocols: centralized and distributed. Centralized approach has a directory or registry, where the different services information (*description, functionality and their address*) are registered in the directory, the service discovery is then provided by sending unicast queries to the directory. Distributed approach demands that objects collaborate and communicate using broadcast or multicast mechanism to discover a service. The IoT services are deployed in resource-limited objects, traditional context-aware web service discovery centralized approaches as Universal Description, Discovery and Integration (UDDI) are not suitable for service discovery in IoT [8, 9] due to the differences between real world services and traditional ones, and it is impossible to register all service providers by different nodes in UDDI [10]. Mobile devices impose limitations to service discovery due to the limited energy, storage capabilities and computing [11]. Therefore, the centralized approach can't ensure scalability in an IoT environment, which is affirmed by several works such as [12][13][14]. In order to remedy to the scalability problem, new protocol recently proposed by AbdelFadeel called 6LoWDIS [15] is built on CoAP/HTTP and requires the support of multicast routing in the network layer by using gateways to discover. However, the use of the gateway may not always be available to assume the service discovery.

A decentralized service discovery approaches have been widely studied in the literature in IoT. Cirani et al [16] propose discovery service for large scale IoT based on P2P technology and Zero configuration, which focuses on both local and global service discovery and used IoT gateway to store information for accessing services, this IoT gateway acts as HTTP-to-CoAP proxy. The gateway keeps track of any things joining or leaving its network and updates the list maintained at its CoAP server. However, this can be insufficient for guaranteeing the availability of published service descriptions. The authors in [17] propose a P2P architecture designed for IoT applications. This approach concerns only the discovery problem proposing a two-tier DHT overlay implemented by the gateways for each IoT network. Therefore, the use of the gateway may not always be available to assume the service discovery [18]. Rapti et al. [19] propose a decentralized service discover based on artificial potential fields (APFs), which are formed upon each user service request and becomes active at points where services can be provided. However, this proposition focuses on local service discovery. In [11], the authors propose the discovery and selection of mobile services using Artificial Neural Networks. However, this discovery mechanism is based only on the pre-defined set of services.

Flooding and random walk approaches have been widely used in communication networks such as peer to peer network because of their good performance [20][21][22][23]. With flooding, a node broadcasts a request to all neighbors, which in their turn broadcast the request to their neighbors until the request results are found. However, this approach suffers from network overheads in search process. To remedy the problem of huge network overheads, Barjini et al. in [24] present flooding-based search techniques in unstructured P2P networks, and consider two variation types, namely, TTL Limit-Based Flooding, and Probabilistic Limit-Based Flooding (PLBF). However, this approach suffers from the scalability problem. Random walk is the improved version of flooding PLBF, where a node forwards the request to only one randomly chosen neighbor at each step until the request results is found. Among emerging standards for decentralized service discovery, Jini technology [1] uses Java Remote Method Invocation protocols [25], Service Location Protocol (SLP) [26], Universal Plug and Play (UPnP) [27] and anther protocol Domain Name System-based Service Discovery (DNS-SD) [28]. SALUTATION is another protocol developed by the Salutation Consortium [29], it

-[1] http://eia.udg.es/~teo/sd/documents/articles/jini2_0.pdf

TABLE I. COMPARATIVE TABLE FOR SERVICE DISCOVERY PROTOCOLS

Protocol	Infrastructure size	Directory architecture	Programming language	Discovring entity	Scalability	Security
Jini [25]	Entreprise network	Distributed	Java	Lokup table	-	Acces contol list
SLP[26]	Entreprise network	Distributed	Independent	Directory Agent (DA)	Used more DA	Integrity
UPnP[27]	Entreprise network	Distributed	Independent	-	-	-
Salutation[29]	Entreprise network	Distributed	Independent	Service broker SLM	-	Authentification
DNS-SD[28]	Entreprise network	Distributed	Independent	-	Multi-link networks	Authentification
Rapti[19]	Iot	Distributed	Independent	-	Yes	-
UDDI [30]	*Arbitrary Network*	Centalized	XML format	UDDI registry	No	Authentication

define an entity called the Salutation Manager (SLM) that works as a service broker for applications, services and devices called Networked Entities. However, these protocols are used in non-constrained networks. A comparison of features of the service discovery protocols are summarized in Table I.

3. PRPOBLEM STATELENT

The heterogeneity of objects, their capabilities (limited energy, storage capabilities, computing power), mobility, properties, different communication technologies, and protocols bring many challenges and requirements to the service discovery. In IoT, there are generally two approaches for the service discovery protocols: centralized and distributed. Most of global discovery mechanisms use a central directory maintained by a gateway to discover. However, this can't guaranteeing the availability and scalability of services and maintaining a single point of failure gateway that is the main drawback of this approach. Researchers addressed this problem by distributed approach with using a number of interconnected cooperating directories. However, this approach is faced with very heterogeneous devices and systems. Other solutions take inspiration from some biological phenomena, offering scalability, fault tolerance, and flexibility in unstructured organizations.

In our solution, we exploit Ant Colony Algorithm in decentralized way, the directory problem was as a multi-agent system, where agent is ant, the ant chooses a promising neighbor, to which it forwards the request. The selection is performed by using a probability value, calculated based on the density of pheromone and heuristic information.

4. ANT COLONY ALGORITHM

The Ant Colony Algorithm metaheuristic is a nature-inspired technique; it is used to find good solutions to difficult discrete optimization problems. Initially, Deneubourg studied the behavior of real ants in search of food [31], Ant colony-based algorithms were originally proposed by Dorigo, Maniezzo and Colorni [32]. They were designed to solve the Traveling Salesman Problem; the first algorithm proposed is Ant Colony System [33]. There are now available numerous successful applications of the ACA metaheuristic that have been proposed such as Quadratic Assignment Problem (QAP) [34], vehicle routing [35], reliability optimization of a series system with multiple-choice and budget constraints [36] and it is also used for solving the new integrated model of job shop scheduling and conflict-free routing of Automated Guided Vehicles [37].

In this paper, ACA is adopted to discover a set of services requested by a user in an IoT network with a totally decentralized approach without intervention of a user. The solution construction process is based on a pheromone deposited on trails to find the short paths between their colony and food sources. Each object in IoT provides set of services become the food sources, the behavior of ant colonies is exploited in this article to discover and to localize a suitable set of services in an IoT network, to satisfy a user's request and to find the solution with a controlled number of hops. In order to limit number of hops, the inverse of distance parameter of ACA is the number of services.

5. Proposed Solution

When a user sends a service request, we apply an Ant Colony Algorithm to find the shortest path with low number of hops required in order to satisfy the request. This algorithm has three steps; the first step is to generate a number of ants. In the second step, we should allow each ant to build a solution, from the initial state (*object sending the request in first time*), then these ants choose the following objects, according to the pheromone concentration and number of services provided by the neighbors, and they check the availability of the services in the request until they form a solution (*all the objects necessary for the satisfaction of the request*). The ants updated the pheromone on the path where the solution is found. In the third step, we choose the best solution, then the ants begin again this process until finding the satisfactory solution. This process is descripted in algorithm 1.

A. Model of Ant Colony Algorithm

As shown in Fig. 2, an IoT network can be defined as a graph G=(W, L), where:

$W=O \cup S$; $O=\{O_1, O_2, \ldots, O_M\}$ is a set of objects and $S=\{S_1, S_2, \ldots S_{Nb\_S}\}$, is a set of possible services provided by objects O_i used generally a wireless links E_{ij}.

$L=\{L_{Oi,Oj} | (O_i, O_j) \in O\}$ is a set of possible links among the smart devices. The fitness function of ACA can be expressed as follows:

$$f(O_i) = \text{Length}(L_i)-1 \qquad (1)$$

The fomula 1 return the number of hops used to discover a service, where L_i is the list of objects used par O_i to discover a service.

We define a set of artificial ants that traverse the IoT network in order to find the set of services requested. In its search, an artificial ant can explore the IoT network, it has to choose an object that links the object where the ant is currently positioned. The choice is stochastically determined with the probability P (see Formula 2) taking into account the density of pheromone τ (see Formula 3) and some other heuristic information on the object leading to a candidate. The heuristic information η refers to the number of services provided by object.

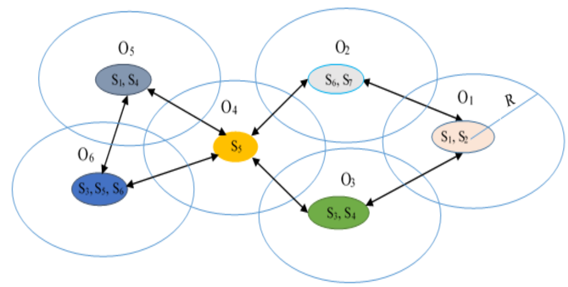


Figure 2. Representation of IoT network

An illustrative example of service discovery protocols used ACA in internet of things is shown in Fig. 2. Here, we consider a network of objects (O_1, O_2, O_3, O_4, O_5, O_6) connected using wireless technology. Each objet provides services $\{S_1, S_2, S_4, S_5, S_6, S_7\}$. The object O_1 sends a request of services $\{S_5, S_6\}$. In order to satisfy the request, the ACA algorithm adapted to the service discovery launches the request on the neighbors $\{O_2, O_3\}$, the neighbors who offer more services and the pheromone density is higher are more likely to be chosen. The pheromone will be deposited each time an ant finds a solution after running the algorithm. The solution converges to the optimal solution with minimum number of hops, it is $\{O_1, O_2, O_4\}$ with two hops.

- **Probability of path selection** $P_{ij}^{k}(t)$

$P_{ij}^{k}(t)$ is defined as the transition probability that the k[th] ant walks forwarded from O_i to O_j at *t* iteration, which can be expressed as follows:

$$P_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{y\in N_i^k}\left(\left[\tau_{iy}(t)\right]^{\alpha}\left[\eta_{iy}\right]^{\beta}\right)} & \text{if } j\in N_i^k \\ 0 & \text{Otherwise}\end{cases} \qquad (2)$$

N_i^k is the set of neighbors non-visited nodes where the k[th] ant is located in node *i*.

$\tau_{ij}(t)$ is the amount pheromone in the link (i, j) at *t* iteration.

η_{iy} is the heuristic information, that is the number of services offered by object *y* using the link (i,y).

α is a parameter to control the degree of importance pheromone in link (i,j).

β is a parameter to control the influence η_{iy}.

- **Update of the Pheromone** After finding a services requested, the density of the pheromone $\tau_{ij}^{k}(t)$ in all paths should be updated at each iteration, as shown in the formula (3)

$$\tau_{ij}(t+1) = \sum_{k=1}^{Nb\_Ants} \Delta\tau_{ij}^{k}(t) \qquad (3)$$

Nb_Ants is the number of ants

$\Delta\tau_{ij}^{k}$ is the incremental of the pheromone density at the t^{th} iteration, which can computed as shown in the formula (4)

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L^{k}} & if\ (i,j) \in path\ \ done\ by\ ant\ \ K \\ 0 & otherwise \end{cases} \qquad (4)$$

L^k is the total length of paths after the k^{th} ant to discover the request of services.

Q is a constant to control the increasing speed of $\Delta\tau_{ij}^{k}$

Ants add pheromones in their path, and the pheromones will evaporate over iteration to avoid the unlimited accumulation of pheromone and allows the algorithm to forget the bad decisions previously taken. The pheromone evaporation formula between O_i and O_j at $(t+1)$ iterations is as follows

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^{Nb\_Ants} \Delta\tau_{ij}^{k} \qquad (5)$$

ρ is a coefficient that represents the degree of evaporation of pheromones, its value is between 0 and 1.

- **Selection of path L^s**

  The selection of the best solution $\mathbf{L^s}$ is constructed by ACA algorithm at *t* iteration, $\mathbf{L^s}$ is the minimum path in number of hops to discover the request of services, the selection is given by formula (6) :

$$L^s = Min\_hops(L^k) \qquad (6)$$

L^k is the path that provide the requested services by ant k / k=1.. Nb_Ants.

B. Algorithm service discovery based on ACA

The proposed lookup algorithm is illustrated in algorithm 1, the different steps of the proposed ACA algorithm are described as follows:

Step 1: The initialization of the parameter; t=1, α β, *Q*, ρ, $\tau_{ij}(0)=c$, Nb_Ants, N.

Step 2: Place the initial requester object in the current solution list L^k.

Step 3: choose the next object with transition probability calculated using the formula 2 and then update L^k.

Step 4: if all services are discovered, then calculate the length L^k (number of hops) that k^{th} ant has passed through, and updated L^s ; L^s is the shortest path in number of hops. Otherwise, go to step 3.

Step 5: calculate $\Delta\tau_{ij}$ using formula 4 and update $\tau_{ij}(t)$ using formula 3.

Step 6: if k<= Nb_Ants, update the information $\tau_{ij}(t)$ using formula 5,then turn to step 2.

Step 7: if t <=N, then turn to Step 1; otherwise, print out the shortest path and then terminate the entire program.

Algorithm 1. ACA-based Lookup discovery mechanism

Begin

Inputs: *Req* (S_i,O_i) S_i is set of requested services by object O_i
1: Initialization:
$\tau_{ij}(0) = c$ for each $e(i,j) \in E,\ i, j \in O,\ i \le M,\ j \le M$
N, α β, Q, ρ, Nb_Ants, L^s=inf;

2: **For** t=1 to N **do** (N=number max of iterations)
3: **For** k=1 to Nb_Ants **do** (Nb_Ants= number of ants)
4: $S_r=S_i$;
5: L^k={i**},** Place the initial requester starting node in the current solution list L^k
6: h=last{ L^k } place the last of list L^k in h
7: V_h is set of neighbour objects $e(h,j) \in E,\ h, j \in O,\ j \le M$.
8: Calculate $P_{ij}^{k}(t)$ $j \in V_h$ by formula 2
9**:** Choose the next object **p** according to transition probability value calculated and append the k^{-th} ant's list
11: $L^k = L^k \cup \{p\}$; **p** $\in V_h$, ns=length(S_r)
12: **For** t=1:ns
13: **If** service $S_r(t)$ provided by P then
14: $S_r = S_r - S_r(t)$;
15: **End If**
16: **End for**
17: **If** $S_r \neq \{\}$
go to 6
18: **Else**
19 Calculate length of L^k
20: **If** length(L^k)< L^s then
21: $L^s = L^k$ updated that to the shortest path.
22: **End If**
23 calculte $\Delta\tau_{ij}^{k}$ and update trails using formula 4
24: update the information $\tau_{ij}(t)$ for each E(i,j) $\in L^k$ using formula 3
25: **End If**
26: **End For**
27: update the information $\tau_{ij}(t)$ using formula 5
28: **End For**
Ouput: L^s the shortest path
End

The flowchart of the algorithm 1 is shown in Fig. 3.

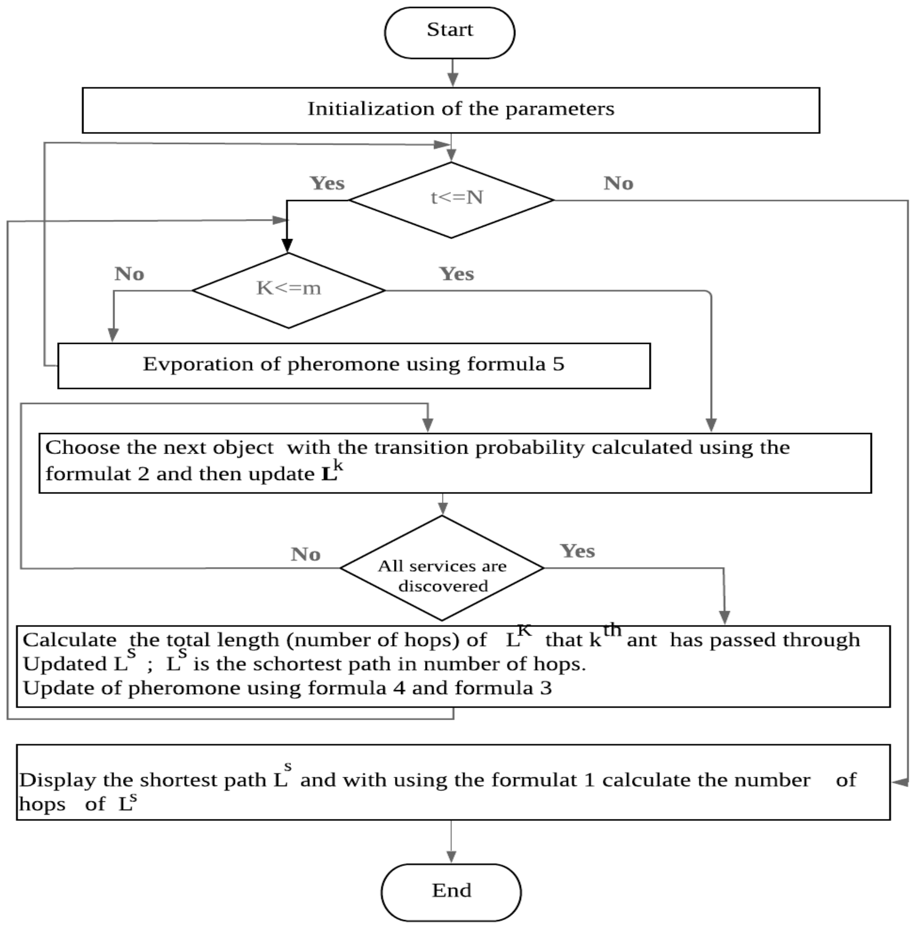


Figure 3. Flowchart of the ACA-based Lookup discovery mechanism

6. PERFORMANCES EVALUATION

A. *Simulation settings*

In order to evaluate the proposed decentralized service discovery for the pervasive environment in IoT networks, the network topology is with uniformly distributed objects. Each object in the IoT network is able to communicate wirelessly with other objects in their proximity and it provides at least one service, and the services are uniformly distributed in each network configuration. Each request of services is randomly associated to an object IoT.

The tests evaluate the performance of the service discovery and localization in the decentralized environment of using ACA algorithm with the parameters described in the table II in order to calculate the average number of hops required to satisfy a request.

B. *ACA parameters*

In order to validate our proposed method, we have implemented our experents in MATLAB R2016b. The proposed ACA algorithm is run on the PC with an Intel Duo 2.4 GHZ CPU and Microsoft Windows 7 enterprise edition SP1 64-bit operating system. Table II describes the simulation parameters with their values. In order to evaluate the influence of the ACA algorithm in different scale environments, the network consists of 50, 100 and 1000 objects respectively. In addition, discovery behavior is tested for queries containing a variable number of different services 3, 5, and 10. The parameter settings for the Ant Colony Algorithm are as follows:

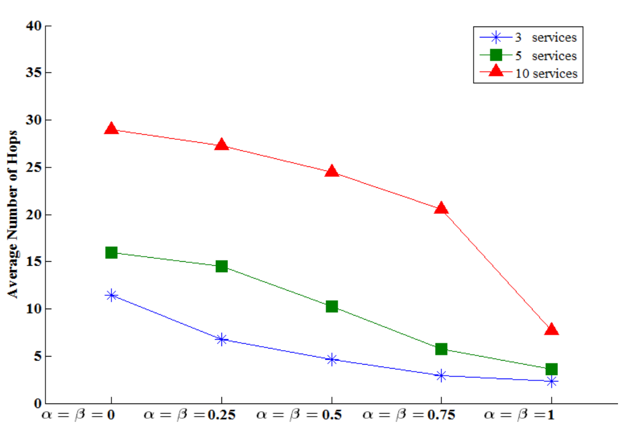
TABLE II. SIMULATION PARAMETERS

Transmission Range, R	25 m
Maximum Iteration, N	100
Number of ants, Nb_Ants	100
Number of services per object	1-3
Q, is a constant	70
ρ, the degree of evaporation pheromones	0.9

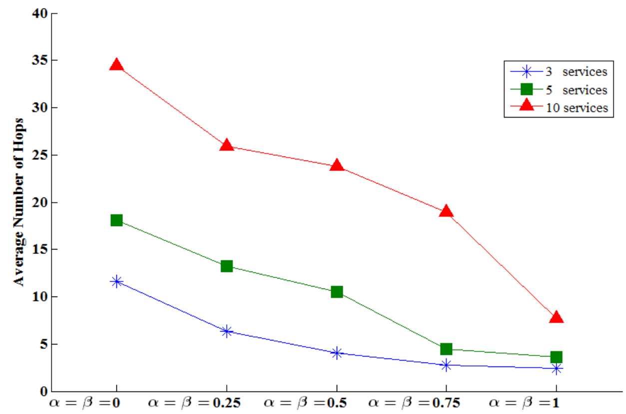
We have chosen the average hop required in order to satisfy a request as metric to evaluate system performance. For each network configuration, 100 simulations were performed and each value in the graph is the average of 100 measurements.

Fig. 4 shows the influence of α and β parameters (*ACA parameters*) on the services discovery in different topology scales (50,100 and 1000), and in addition, the discovery tested on requests that contain (3,5 or 10) services, while counting the number of hops required to satisfy the request. The heuristic information and Information elicitation have no effect on path discovery. To analyze the influence of the parameters of the ACA algorithm in the service discovery proposed in this article, the parameters α and β took different values in the interval [0, 1]. The pheromone increasing strength coefficient *Q* is 70. The value of the pheromone evaporation coefficient Rho is 0.9.

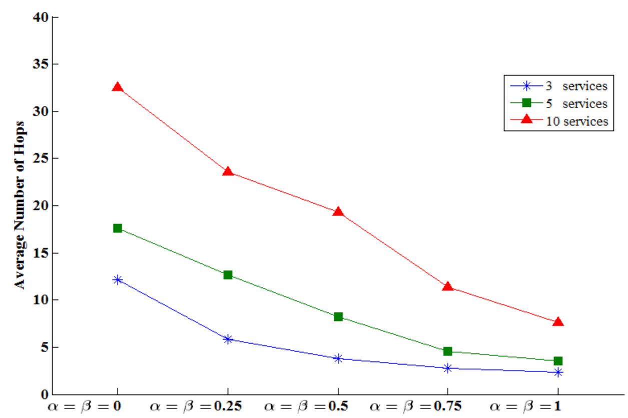
A value of α and β that is equal to 0 means that the discovery is similar to a random search, in this case, ACA found a solution with a higher number of hops that is required to satisfy a request containing a different number of services.



a) 50 objects



b) 100 objects



c) 1000 objects

Figure 4 Average number of hops required for the discovery services for different *α* and β values.

C. Routing metric and scalability

In order to validate our approach, we compared the obtained results in different scale of topology (50,100,500, 1000) with a request that contains 5 services with flooding-based and random walk-based approaches. The routing metric is the number of hops required to satisfy a request. Fig. 5 shows that the flooding-based approach requires a lower number of hops for satisfied a request. However, the difference in the performance with our approach is less than one hop for each network configuration. Without ignoring the huge number of messages exchanged between objects generated by flood-based approaches in such random topology networks, this causes big control overhead, and thus leading to a much higher energy consumption. which could introduce a significant cost for the resource constrained devices of IoT networks. On the other hand, the pure random walk-based discovery, due to stochastic behavior.

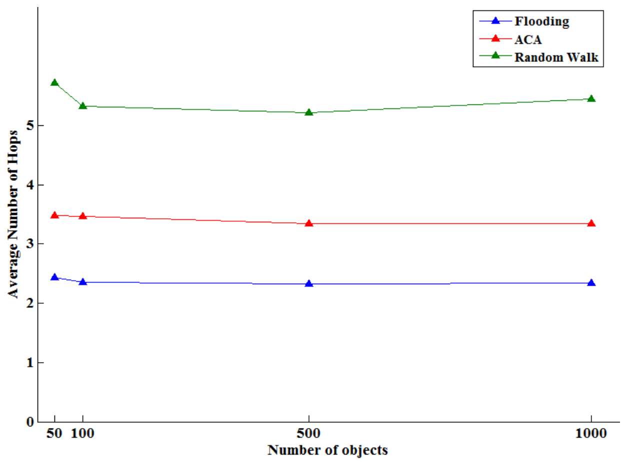


Figure 5. Average number of hops required for the discovery services for ACA-based and flooding-based approaches in different size networks.

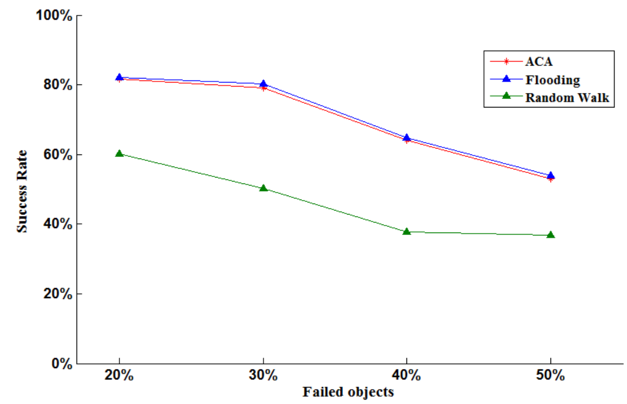
According to the above analysis, it can be concluded that the best configuration is obtained when alpha and beta take the value 1. The request is satisfied with a sufficiently small number of hops.

D. Success Rate

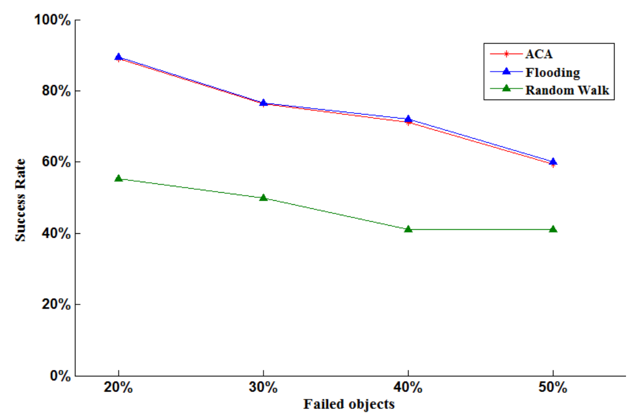
The success rate is defined as percentage of the number of successful discovery requests, out of the total number of discovery requests. To control the numbers of hops, we added the Time-To-Live (TTL), the TTL in our simulation specifies the number of hops that can be used to discover the request. The TTL initialized to 5. The TTL is decreased at each visit of an object, when TTL equals to 0 and the request does not satisfy the request considered failed.

In order to evaluate the success rate of the ACA algorithm in the service discovery proposed in this article, simulation experiments were conducted and compared with flooding-based and random walk approaches. For each simulation, 100 requests to discover, the requests contain 5 services in different scale environments, the network consists of 50 and 100 objects. The percentage of failed objects took different values [20%, 30%, 40%, 50%].

From the Fig. 6, which correspond to networks content respectively 50 and 100 objects, we can conclude that our approach has a similar behavior to flooding in the number of satisfied requests and this varies according to the number of failed objects. In addition, the flooding approach uses the broadcast scheme to the discovery request to all the neighboring objects. This process iterates until the desired target objects is found, this causes big control overhead, and thus leading to a much higher energy consumption. Which could introduce a significant cost for the resource constrained devices of IoT networks. The random walk algorithm has very poor performance, due to the random behavior, and the success of the discovery depends on the choice of a sufficiently long TTL value in this case it has small value (TTL=5).



a) 50 objects



b) 100 objects

Figure 6. Success Rate for services discovery

7. CONCLUSION

In this paper, an approach for decentralized services discovery and localization in IoT network is proposed; it is based on an Ant Colony Algorithm. This algorithm consists on using the pheromone as a learning mechanism allowing the colony to converge towards solutions, which is based on the deposition of traces pheromones. These traces pheromone used to skew decisions when building solutions. In IoT network, each object provides services. To satisfy a received request, it is shown how ACA is adopted to solve the problem of discovery and localization of services in a network.

This algorithm can achieve a high success rate in service discovery and finding the best solution with an optimized number of hops and has a similar behavior despite the increase in the size of the network.

As a future works, our research will focus on the integration of the semantic aspect in the service discovery and we think it will be very beneficial for IoT networks with heterogeneous objects.

REFERENCES

[1] CISCO: “Cisco Visual Networking Index: Global IP Traffic Forecast, 2014–2019”. Tech. rep.,CISCO, 2015

[2] D. M. Kumar, and A. Ghosh, “Resource Efficient Routing in Internet of Things: Concept, Challenges, and Future Directions”, *International Journal of Computing and Digital Systems*, *8*(6), 2019.

[3] L. F. M . Vieira, U. Lee, and M. Gerla, “Phero-trail: a bio-inspired location service for mobile underwater sensor networks”, *IEEE Journal on Selected Areas in Communications*, *28*(4), 553-563, 2010

[4] S. L. Chao, H. Y. Lee, C. C. Chou and H.Y. Wei, “Bio-inspired proximity discovery and synchronization for D2D communications”, *IEEE communications letters*, *17*(12), 2300-2303, 2013.

[5] M. Premkumar, A. M. Ibrahim, R. M. Kumar and R. Sowmya, “Analysis and Simulation of Bio-Inspired Intelligent Salp Swarm MPPT Method for the PV Systems under Partial Shaded Conditions”, *International Journal of Computing and Digital Systems*, *8*(5), 490-496, 2019

[6] L. Atzori, A. Iera and G. Morabit, “Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm”. *Ad Hoc Networks*, *56*, 122-140, 2017

[7] B. Djamaa, M. Richardson, N. Aouf and B. Walters, “Service discovery in 6LoWPANs: classification and challenges”, in *2014 IEEE 8th International Symposium on Service Oriented System Engineering* (pp. 160-161). IEEE, 2014, April 2014.

[8] Q. Wei, Z. Jin. “Service discovery for Internet of Things: A context wareness perspective”, Proceedings of the FourthAsia-Pacific Symposium on Internet ware. ACM. 2012. pp.25.

[9] B. C. Villaverde, R. D. P Alberola, A. J. Jara, S. Fedor, S. K. Das, D. Pesch., “Service discovery protocols for constrained machine-to-machine communications”. IEEE Commun. Surv. Tutor. **16**(1), 41–60, (2014).

[10] B. Jia, W. Li and T. Zhou, “A novel P2P service discovery algorithm based on Markov in internet of things”, in the 2017 I.E. International Conference on Computational Science and Engineering (CSE) and Embedded and Ubiquitous Computing (EUC), Guangzhou, China, pp. 26–31, Vol. 2. IEEE,2017

[11] E. Al-Masri, Q. H. Mahmoud, "A context-aware mobile service discovery and selection mechanism using artificial neural networks”, in: Proceedings of the 8th international conference on electronic commerce: the new e-commerce: innovations for conquering current barriers, obstacles and limitations to conducting successful business on the internet. ACM, pp. 594–598,2006.

[12] G. Cassar, P. Barnaghi, W. Wang, K. Moessner, “A Hybrid Semantic Matchmaker for IoT Services.” in Internet of Things (iThings), IEEE International Conference. IEEE.2012.pp. 210-216. 2012

[13] W. Wang, S. De, G. Cassar, and K. Moessner, “Knowledge representation in the internet of things: semantic modelling and its applications”, *automatika journal*, *54*(4), 388-400,2013.

[14] Q. He, J. Yan, Y. Yang, R. Kowalczyk, and H. Jin, “A decentralized service discovery approach on peer-to-peer networks”, *IEEE Transactions on Services Computing*, *6*(1), 64-75,2013.

[15] K. Q. AbdelFadeel and K. Elsayed, “6LoWDIS: A lightweight service discovery protocol for 6LoWPAN”, in *2016 IEEE International Conference on Communications Workshops (ICC)* (pp. 284-289). IEEE, may 2016.

[16] S. Cirani, L. Davoli, G. Ferrari, R. Léone, P. Medagliani, M. Picone and L. Veltri, “A scalable and self-configuring architecture for service discovery in the internet of things”, *IEEE Internet of Things Journal*, *1*(5), 508-521,2014

[17] G. Tanganelli, C. Vallati and E. Mingozzi, “Edge-centric distributed discovery and access in the internet of things”, *IEEE Internet of Things Journal*, *5*(1), 425-438,2018

[18] J. Quevedo, C. Guimarães, R. Ferreira, D. Corujo and R.L. Aguiar, “Icn as network infrastructure for multi-sensory devices: Local domain service discovery for environments”. *Wireless Personal Communications*, *95*(1), 7-26, 2017.

[19] E. Rapti, A. Karageorgos, C. Houstis and E. Houstis, “Decentralized service discovery and selection in Internet of Things applications based on artificial potential fields”, *Service Oriented Computing and Applications*, *11*(1), 75-86, 2017

[20] A. O. Stauffer, V.C. Barbosa, “Probabilistic heuristics for disseminating information in networks”, IEEE Trans Netw 15:425–435,2007.

[21] N. B. Chang and M. Liu, “Controlled flooding search in a large network”, *IEEE/ACM Transactions on Networking (TON)*, *15*(2), 436-449, 2007.

[22] R.Gta and M. Sereno, “Generalized probabilistic flooding in unstructured peer-to-peer networks”. IEEE Transactions on Parallel and Distributed Systems, 22(12), 2055-2062, 2011.

[23] F. Palmieri, “Scalable service discovery in ubiquitous and pervasive computing architectures: A percolation-driven approach”, Future Generation Computer Systems, 29(3), 693-703, 2013.

[24] H. Barjini, M.Othman, H. Ibrahim and N. I. Udzir., “Shortcoming, problems and analytical comparison for flooding-based search techniques in unstructured P2P networks”, Peer-to-Peer Networking and Applications, 5(1), 1-13,2012.

[25] Sun microsystems, jini architecture specification version 2.0http://eia.udg.es/~teo/sd/documents/articles/jini2_0.pdf,2003.

[26] E. Guttman, C. Perkins, J. Veizades and M. Day, *Service location protocol, version 2* (No. RFC 2608). Available: https://tools.ietf.org/html/rfc2608,1999.

[27] U. Plug, “Play (UPnP) Forum”. *Microsoft Corporation,2003.*

[28] S. Cheshire and M. Krochmal, “*DNS-Based Service Discovery*”,Internet Engineering Task Force, RFC 6763 [Online]. Available: http://tools.ietf.org/html/rfc6763 42. Universal Plug (2003) Play (upnp) forum. Microsoft Corporation

[29] Salutation Architecture Specification Version 2.0c - Part 2, The Salutation Consorium, June 1, 1999. http://www.salutation.org

[30] T. Bellwood, D. Bryan, V. Draluk, D. Ehnebuske,T. Glover and A. Hately, “UDDI version 2.04 API specification”, *UDDI Committee Specification, OASIS*, 2002.

[31] J. L. Deneubourg, J. M. Pasteels and J. C. Verhaeghe, “Probabilistic behaviour in ants: a strategy of errors?”, *Journal of Theoretical biology*, *105*(2), 1983.

[32] M. Dorigo, V. Maniezzo and A. Colorni, “The ant system: An autocatalytic optimizing process.59-271, 1991.

[33] M. Dorigo and L. M. Gambardella, “Ant Colony System : A Cooperative Learning Approach to the Traveling Salesman Problem”, IEEE Transactions on Evolutionary Computation, 1(1), 53_66, 1997

[34] L. Gambardella, E. Taillard and M. Dorigo, "Ant Colonies for the Quadratic Assignment Problem", Journal of the Operational Research Society, 50, 167_176, 1999.

[35] B. Bullnheimer, R. F. Hartl, R. F and C. Strauss, "An improved ant System algorithm for thevehicle Routing Problem". *Annals of operations research*, *89*, 319-328, 1999.

[36] N. Nahas and M. Nourelfath, "Ant system for reliability optimization of a series system with multiple-choice and budget constraints". *Reliability Engineering & System Safety*, *87*(1), 1-12, 2005

[37] M. Saidi-Mehrabad, S. Dehnavi-Arani, F. Evazabadian and V. Mahmoodian, "An Ant Colony Algorithm (ACA) for solving the new integrated model of job shop scheduling and conflict-free routing of AGVs". *Computers & Industrial Engineering*, *86*, 2-13, 2015.

Zahia BENSALAH AZIZOU is currently a Ph.D. student at LIMPAF laboratory, Bouira University, Algeria. She received the engineer degree in 2006 in Computer Science (Artificial intelligence option) from the University of Bejaïa (Algeria), the Magister degree in 2010 in Computer Science (Networking and Distributed Systems option) at the University of Béjaïa (Algeria), she works as a teacher at Béjaia University. Her research interests are: service discovery in Internet of Things, Modeling, Simulation, Performance evaluation, and Quality of service.

Abdelmalek Boudries is an associate professor at Bejaia University. He is member of LMA Laboratory (Laboratory of Applied Mathematics). He received the engineer degree from the University of Tizi-Ouzou (Algeria) in 2000 and the Magister degree from the University of Bejaia (Algeria) in 2007. Since June 2015, he is a PhD at the University of Bejaia. In May 2017, he obtained the HDR in computer sciences from Setif University (Algeria). His main research interests ad hoc and wireless sensor networks (*Routing*, *connectivity maintenance*,...), Social Networks, Internet of Things, Web Services, Business Intelligence, Articial Intelligence, ...

Mourad Amad is an associate professor at Bouira University. He received the engineer degree from the National Institute of Computer Science (*INI-Algeria*) in 2003 and the magister degree from the University of Bejaia (*Algeria*) in 2005. Since January 2012, he is a PhD at the University of Bejaia, In May 2016; he obtained the HDR in computer sciences from Bejaia University. His research interests include: Peer to peer networks (*Lookup acceleration, Security, Performance evaluation, Mathematiques Modelling, ...*), Sensor Networks (*Clusterisation, Routing, ...*), Media Conferencing (*architecture, ALM, signalling protocol, P2P-SIP, ...*), Social Networks, Internet of Things (*Architecture, Fault Tolerance,*)