

A Diffusion-Based ACO Resource Discovery Framework for Dynamic p2p Networks

Kamil Krynicki

ISSI – Departamento de Sistemas
Informáticos y Computación,
Universitat Politècnica de València
Valencia, Spain
kkrynicki@dsic.upv.es

Javier Jaén

ISSI – Departamento de Sistemas
Informáticos y Computación,
Universitat Politècnica de València
Valencia, Spain
fjaen@dsic.upv.es

Alejandro Catalá

ISSI – Departamento de Sistemas
Informáticos y Computación,
Universitat Politècnica de València
Valencia, Spain
acatala@dsic.upv.es

Abstract — The Ant Colony Optimization (ACO) has been a very resourceful metaheuristic over the past decade and it has been successfully used to approximately solve many static NP-Hard problems. There is a limit, however, of its applicability in the field of p2p networks; derived from the fact that such networks have the potential to evolve constantly and at a high pace, rendering the already-established results useless. In this paper we approach the problem by proposing a generic knowledge diffusion mechanism that extends the classical ACO paradigm to better deal with the p2p's dynamic nature. Focusing initially on the appearance of new resources in the network we have shown that it is possible to increase the efficiency of ant routing by a significant margin.

Keywords—Ant Colony Optimization, ACO, p2p, Semantic Search, Semantic Networks, Overlay Networks

I. INTRODUCTION

Since the introduction of the ACO metaheuristic [1] it has been considered a dynamic algorithm, one that is capable of successfully adapting in case a change in the system occurs [2], very much resembling the actual behavior of ants. One can indicate however a pace of the evolution of the underlying structure at which the classical ACO will not be able to keep up and might achieve highly suboptimal results during the entirety of its execution; we can view it as a certain level of inertia of the system. Naturally, ACO thrives at solving static problems; both: classical, such as the Traveling Salesman Problem [3] and the Assignment Problem [4], but also more uncommon ones, as shown in our several works [5] [6] [7]. Nevertheless, the dynamic issue has remained largely unaddressed. In various papers [8] [9] [10] it has also been shown that ACO-based algorithms are more than an adequate base for a p2p search engine. However, p2p networks are innately and unpredictably dynamic which makes the mentioned problem relevant.

As indicated in [11] the dynamism of a network can impair severally the quality of the results obtained. The issue centers on the fact that ACO algorithms are not capable of redirecting the established routes immediately; a problem to which we refer as slow re-convergence. In a moderate size p2p network it can take up to a few thousand execution iterations in order to accommodate the changes, in this time an additional step of the system's evolution might occur and make a full convergence nearly impossible. In order to counteract this problem we introduce a diffusion layer in p2p networks, within the scope of ACO's pheromone paradigm, designed to propagate the

information quicker and more efficiently than a pure ACO would. We named it the Diffusion Model Framework.

Classically the ants were considered the only 'intelligent' element in the system, while the rest was purely passive, only capable of receiving, resending and answering queries. In our case we let the nodes themselves take some of the responsibility, by being able to analyze its own content and take decisions, thus redistributing the focus more in the network.

In this paper we introduce a novelty in the field of p2p semantic search, namely our Diffusion Model Framework as an entire platform that permits defining many diffusion algorithms of diverse nature. The main idea behind it is to improve the re-convergence of the system to a degree that allows an efficient p2p search, regardless of the level of the system's dynamism. In this respect, we propose two diffusion strategies: in-width and in-depth diffusion, both built within the mentioned framework and we perform an experimental study of these strategies to evaluate their effectiveness. We chose the Ant Colony System (ACS) [1] as our ACO implementation, seeing that it is considered the most classical of all, with one crucial modification called the Routing Concept (RC), which was proposed, in different aspects, by several authors [12] [8]. The results obtained by the above studies show that RC can significantly improve query routing in ACO-based p2p systems [10].

The experimental data obtained demonstrate that, in terms of re-convergence speed, the in-width strategy has proven to be a feasible solution in systems that have a downtime period. Moreover, in the case of the in-depth diffusion, the overall strain on the system is minimal and the effectiveness of the search process is improved up to 30% with respect to a non-diffusion-based search process.

The paper is structured as follows. In section II we summarize the related work in the field of p2p and ACO with respect to information diffusion. In section III we introduce some of the ACS basis and expand on the subject of the Routing Concept. In section IV we describe in detail the Diffusion Model Framework and the concrete implementations we chose to experiment with. In section 0 we present our experimental setup, formalize all the concepts and propose a quality measure; section VI shows the results obtained and in section 0 we conclude and summarize our main findings.

II. RELATED WORK

To the best of our knowledge the subject, as we define it, has not been explored. There are several works that focus on the diffusion of the resources in the system, but in all the cases it is solely a study of how it occurs naturally in various systems [13] [14], regardless whether they include ACO concepts or not. Never is it an induced diffusion, with the objective of improving a specific quality measure nor is it an extension of the ACO paradigm.

The most similar subject is the publish/feed paradigm where a particular node is the generator of content while a certain subset of the network consumes the resources [15]. However the main focus in those works is the efficiency and scalability, not an ACO-based method nor a resource-query mechanism. We might qualify it as a deterministic and complete point-to-point resource propagation, in contrast to our dynamic resource Diffusion Model Framework.

III. FORMAL BASIS

A. ACS

Ant Colony System [9] is one of the most popular implementations of the ACO metaheuristic. It is an extension and improvement over the Ant System (AS) [16] that we chose as our main strategy for p2p search. The reasons for this choice are twofold: firstly, we wanted to experiment with a fairly well known and pure ACO implementation, secondly, as shown in [11], many of the common ACO implementations are not suitable for neither the query-resource paradigm, the p2p environment nor the dynamic system setup. ACS performed well in all those aspects.

We choose precisely the same mathematical basis of ACS as were presented in [11] [8]. Having this ant strategy in place, we will formalize the semantic search in p2p in a simple manner. Firstly we establish a network of homogenous nodes. Each of them may be in possession of a certain set of resources of any nature, but also each is capable of generating a query – i.e. require resources of a given taxonomical kind. An ant corresponding to the query is created and routed according to the ACO algorithm of our choosing collecting resources labeled with the indicated taxonomical entity from the nodes it visits. Once the algorithm is complete the ant evaluates its findings and returns to its emitting node, performing pheromone-related tasks along the way.

B. Routing Concept

In most ACO algorithms there is just a single value of pheromone per outgoing link in each node. The Routing Concept, however, introduces an additional dimension (a layer), giving each link a full table of pheromones, each corresponding to a class of queries that might appear in a given node. Therefore a single ant, at its creation, is given a Routing Concept value and only manages pheromone that has been deposited in that particular layer of the system. Hence, it is closely related to the Taxonomy Based Routing, presented in [10] [17].

The technical basis is as follows (see [11] for a detailed formal description). Every node N keeps a 2-dimensional matrix $\Omega_N: L_N \times C_N$, with real, positive values, where L_N is the

space of outgoing links from the node N and C_N is the space of Routing Concepts maintained by this particular node N . This matrix is referred to as routing table, or routing matrix. The l -th, c -th element of Ω_N corresponds to the pheromone value of the l -th outgoing link for the c -th routing concept, which can be written as $\Omega_N(l, c) = \tau$.

IV. DIFFUSION MODEL FRAMEWORK

The diffusion is an additional source of pheromone trail creation in the system, which is managed by the nodes themselves and can be performed at will, usually as a response to a node's internal event, such as resource addition. It is achieved through the introduction of a new type of ant that exists alongside the classical Forward Ants (FA) and Backward Ants (BA), namely a Diffusion Ant (DA). Every node of the system is given the ability to analyze its own content's evolution and generate a DA at any moment it sees fit. It can also be externally forced to do so. Each DA carries two values: Diffusion Power and Diffusion Routing Concept. See Table II for details on both of these values; they are established by the emitting node at DA's creation.

TABLE I GLOBAL DIFFUSION PARAMETERS

Parameter Name (symbol)	Parameter Interpretation	Value Constraints
Diffusion Ant Max TTL (TTL_{max})	The maximum distance from the origin node the DA can reach	$TTL_{max} > 0$
Diffusion Delta (D_δ)	The effect of a DA on the node's pheromone	$D_\delta \in \langle 0,1 \rangle$
Diffusion Random Spread Chance (D_ρ)	Chance to continue spreading pheromone to uninitialized regions of the system	$D_\rho \in \langle 0,1 \rangle$
Diffusion Minimum Link Spread (MIN_ρ)	The minimum amount of links to spread to, at each step.	$MIN_\rho \in \{-1, 0, 1, \dots\}$, -1 means all available.
Diffusion Maximum Link Spread (MAX_ρ)	The maximum amount of links to spread to, at each step.	$MAX_\rho \in \{-1, 0, 1, \dots\}$, -1 means all available.

TABLE II LOCAL DIFFUSION PARAMETERS

Parameter Name (symbol)	Parameter Interpretation	Value Constraints
Diffusion Ant Time To Live (DA_{TTL})	The current time to live of a given DA, decreases as an ant continues	$DA_{TTL} \in \langle 0, TTL_{max} \rangle$
Diffusion Power (DA_p)	The scaling parameter of a given DA, affects the pheromone deposition.	$DA_p > 0$
Diffusion Routing Concept (DA_c)	The Diffusion objective of a given DA, affects the pheromone deposition.	One of the Routing Concepts present in the system.

The DA behavior is governed by the below algorithm.

Algorithm: DA behavior

1. DA arrives in node N , DA's Time To Live is DA_{TTL}
2. If Ω_N matrix contains pheromone for the Diffusion Routing Concept carried by DA go to 4
3. Otherwise generate a random real value $s \in \langle 0,1 \rangle$. If $s < D_\rho$ go to 4, else end Algorithm. D_ρ is the Diffusion Random Spread Chance, see Table I for details.
4. Perform Diffusion Pheromone Update (see next algorithm) in N , unless N is the emitting node of DA.

5. If $DA_{TTL} \leq 0$ end Algorithm.
6. Let $aLinks$ be the set of all the outgoing links from N , to nodes not visited previously by the DA nor its clones; let $gLinks$ be a random subset of the $aLinks$ set, of size m , where m is a random integer value, and $m \in \langle MIN_\rho, MAX_\rho \rangle$, limited by the size of $aLinks$
7. Set decrement DA_{TTL} to $DA_{TTL} - 1$ and send a clone of DA to every link in $gLinks$.

Algorithm End

Algorithm: Diffusion Phomone Update in Node N

1. Let

$$\delta\tau(N, N_{in}) = DA_p \times \frac{DA_{TTL} + 1}{TTL_{max} + 1}. \quad (IV.1)$$

where N is the current node, N_{in} is the node from which DA arrived to N and DA_p is the diffusion power.

2. If Ω_N matrix contains phomone for the Diffusion Routing Concept carried by DA (DA_c) go to 5.
3. Add the DA_c to Ω_N .
4. Set the DA_c phomone value, for the link from which DA arrived, at

$$\tau(N, N_{in}) \leftarrow \delta\tau(N, N_{in}) \quad (IV.2)$$

5. In the node N set the phomone value corresponding to DA_c for the link from which DA arrived in N , at

$$\tau(N, N_{in}) \leftarrow D_\delta \times (\delta\tau(N, N_{in}) + p_{init}) + (1 - D_\delta) \times \tau(N, N_{in}), \quad (IV.3)$$

where p_{init} is the default value of the phomone and D_δ is the Diffusion Delta.

Algorithm End

In Table I we summarize the parameters that define the character of the diffusion in question; they remain unchanged during the entire execution of the algorithm. In Table II we summarize parameters attached to a particular act of diffusion, all of which may be variable during a single execution. This choice of parameters allows us to create an abundant amount of types of diffusion. In order to make the analysis more focused we decided to extract two diffusion families with a manageable set of independent variables to examine.

A. In-width diffusion

TABLE III IN-WIDTH DIFFUSION PARAMETERS

Parameter Name (symbol)	Value
Diffusion Ant Max TTL (TTL_{max})	Subject to test: {1, 2, 3, 4}
Diffusion Delta (D_δ)	0.1
Diffusion Random Spread Chance (D_ρ)	1
Diffusion Minimum Link Spread (MIN_ρ)	-1
Diffusion Maximum Link Spread (MAX_ρ)	-1

The most straightforward diffusion is a k-flood diffusion. In simple words it can be understood as unconditionally notifying all the nodes up to the distance k from the initiating node [18]. The k is simply the TTL_{max} in our setup, and therefore the, so called, diffusion depth, would be TTL_{max} as well. Table III presents the values of the diffusion parameters that are necessary to achieve a proper k-flood. See Fig. 1 for a graphical representation of the in-width diffusion and how it affects its neighbors. It creates an area in the p2p network with

increasingly attractive force to the emitter, somewhat similar to a gravity field.

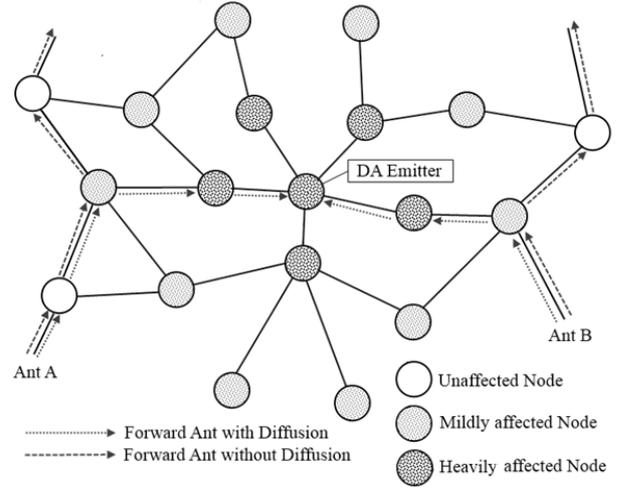


Fig. 1. In-width diffusion, $DA_{TTL} = 2$

B. In-depth diffusion

The opposite approach is the in-depth diffusion. Here we do not permit the cloning of the DA, by establishing $MIN_\rho = MAX_\rho = 1$, which, in combination with the prohibition of the reentry, creates long chains of diffused phomone, rather than areas. The effect DAs have on the system is much lower and more distributed in time, but it is not as strong as the previous method, due to the fact that there might be an ant that misses the chain. As it can be seen in Fig. 2 Ant B behaves identically, regardless of the state of the diffusion. In spite of the given counter-example we suspect that this lightweight mode of diffusion might prove better in terms of the ratio: nodes affected by diffusion to ants attracted; especially so if one chooses to release several independent DA instead of one. Consult Table IV for parameter values necessary to achieve in-depth diffusion in our model. Again, the TTL_{max} will be referred to as the Diffusion Depth later on.

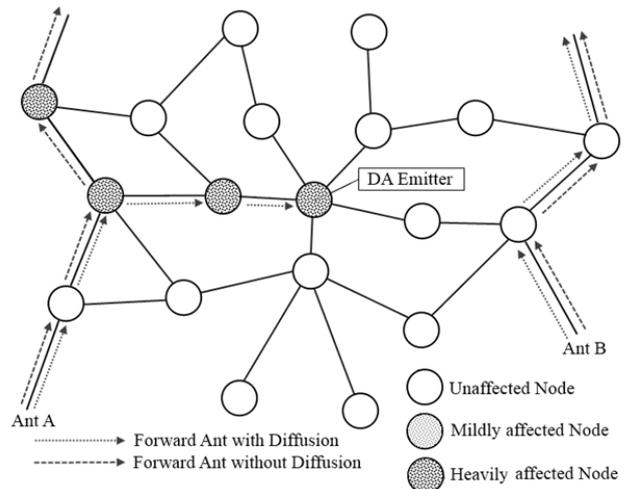


Fig. 2. In-depth diffusion, $DA_{TTL} > 3$

TABLE IV IN-DEPTH DIFFUSION PARAMETERS

Parameter Name (symbol)	Value
Diffusion Ant Max TTL (TTL_{max})	Subject to test: {5,10,15,20}
Diffusion Delta (D_δ)	0.1
Diffusion Random Spread Chance (D_ρ)	1
Diffusion Minimum Link Spread (MIN_ρ)	1
Diffusion Maximum Link Spread (MAX_ρ)	1

V. EXPERIMENTAL SETUP

In this section we provide an exhaustive overview of all the decisions and assumptions we took in order to formalize the experiments.

A. Resource distribution and labeling

As in [10], the ACM Computing Classification System [19] will be the taxonomical vocabulary used. Every resource in the network is described by one, and only one leaf taxonomical concept t (referred to as the taxonomical entity) of the ACM classification. A resource has therefore only two properties – its owner node N and a taxonomical label t . It is depicted as $r(N, t)$. It must be pointed out that two resources $r_1(N_1, t_1)$ and $r_2(N_2, t_2)$, unless explicitly $r_1 = r_2$, are not considered equal, even if $N_1 = N_2$ and $t_1 = t_2$. The consequence of such an approach leads to valuing higher those nodes that provide many resources of the same t .

The distribution of resources within the network follows strictly the approach by the test setup [10]. The resources are evenly distributed among the nodes, as well as among the entities in the taxonomy tree. Additionally, every node is a designated expert in a given field (there can be multiple experts in each field) which is expressed by the composition of resources in it. Of all the resource units in a node, 60% is labeled with the field in which the node is considered an “expert”, further 20% is labeled with another field that is closely related in the taxonomical tree to the expert field, and the last 20% is purely random, but with the restriction to be outside the expert field. This is said to resemble real-world distribution, reflecting the fact that people have specific interests and hobbies [20].

B. Query and query resolution

Every non-diffusion ant is connected to a query q and will only carry one of the ACM classification leaf entities and it will be fully defined by it. In this case, however, $q_1(t_1) = q_2(t_2)$ iff $t_1 = t_2$; the benefit of such an approach is to be able to compare results of two queries released at different time points in the testing process and to show relative improvement between them. The resolution of a query $q(t)$ in a node N_i consists of finding the resources that have been labeled with t , that is, all the resources $\{r \mid \exists r(N_i, t)\}$.

C. P2P network setup

Every p2p network in our tests is a fully connected toroidal world of $N \times N$ nodes, with the resource distribution that obeys the rules explained in section V.A. See Table V for all the world variants. In order to follow a coherent approach with our previous works we chose to present a minor alteration to the pure toroid topology. Namely: every node will additionally have one long distance connection between itself and a node

randomly selected of all those not directly neighboring. The probability of linking two nodes is inversely proportional to the distance between them.

D. Quality measure

As many works before [8] [10] [21] we decided to adopt the Hop per Hit (HpH, dimensionless) measure. It can be read as: how many steps a single ant must take to obtain a single resource during a given iteration. The graphical analysis of many plotted evolutions of HpH in function of iterations led us to confirm that it gets fairly stable after a certain point, called the convergence point [8]. Therefore we chose our final measure to be the arithmetical average of exactly 1000 last values of HpH in a given experiment; we expect it to be characterized by low standard deviation.

TABLE V P2P WORLDS PRESENT IN THE EXPERIMENTS

World edge size	World total size	Initial resources per node	Nodes affected
N	$N \times N$	-	M
32	1024	30	48
40	1600	30	75
48	2304	30	108
56	3136	30	147
64	4096	30	192

Additionally we decided to distinguish between two separate interpretations of the data. It is obvious that every act of diffusion consumes a portion of system resources, whether it is bandwidth, processing time or any other. One might argue, however, that the entire diffusion process can be executed when the system is in its downtime, when there is no or very little real usage. A good example is forcing all the diffusion at night, reducing the probability of a conflict with a user-released ant to a minimum. Hence, in that case the diffusion would not interfere with system’s functionality and it could be omitted in the calculations. Therefore the two interpretations are:

- **non-strained results**, where the resources consumed by diffusion ants are not taken in consideration, and are treated as neutral to the system. The non-strained average of HpH Diffusion Depth d will be symbolized by $\bar{\mu}_d$
- **strained results**, where every diffusion ant is treated as inseparable from the system and included in the results. The strained average of HpH of the Diffusion Depth d will be symbolized by μ_d

A way to include the diffusion strain in the calculations is based around the fact that it does not evolve with the system. We can infer from this that the density of resources consumed by diffusion in the function of iterations remains constant as well. Consequently we sum the total diffusion strain over all the iterations, divide it over the number of iterations and add the resulting value to all the HpH results obtained the classical way, might it be a single iteration or an average of a given period.

E. Experimental methodology

In its most general form the execution of every experiment will be as follows:

Algorithm: Single Experiment in the world $N \times N$

1. Iterate points 2-5 exactly i_{max} times.

2. Choose a random node N_1 of all available in $N \times N$ and select a random resource R of its repository.
3. Choose a random node N_2 of all available in $N \times N$ and generate in it an Ant whose objective is the discovery of the resource R .
4. After its completion the query will report the amount of resources obtained and the amount of steps taken. Data will be used to calculate the HpH measure for that particular iteration.
5. Every $nMod$ iterations initiate the Resource Distribution procedure (see next algorithm).

Algorithm End.

Algorithm: Resource Distribution in the world $N \times N$

1. Choose a set M of random nodes of available in $N \times N$.
2. For a given node $m \in M$, select one random resource from all the ones it possesses and generate a set of 10 resources within its category.
3. Add all generated resources to the node m ; the process referred to as node enrichment.
4. Force the node m to release a diffusion ant into the system that informs about its newly acquired content.
5. Repeat points 2 – 4 for every $m \in M$

Algorithm End

See Table V for sizes of M ; it was adjusted to the scale linearly with the total size of the world. The amount of iterations per execution was chosen to be $i_{max} = 100000$. This particular value permits us to see the full evolution of the system while still presenting us with a manageable amount of data. The period between consecutive resource distributions is set at $nMod = 1000$, therefore the resource distribution will be executed 99 times within a single experiment. The two above statements are concluded from [11].

For each combination of any set of the parameters chosen to test, we will execute the Single Experiment algorithm three times and, after confirming consistency of the results, proceed to work with the average of the three for further analysis.

VI. EXPERIMENTAL STUDY

We decided to formulate the statistical analysis in the following way. Our null hypothesis H_0 is such that the averages of HpH, grouped by a particular Diffusion Depth, in all experiments are equal: $H_0: \mu_1^* = \mu_2^* = \dots = \mu_k^*$, where μ_d^* is the normalized average of HpH values within the Diffusion Depth d . The normalization is with regard to the Diffusion Depth 0; a way to express the relative improvement. The alternative hypothesis H_1 assumes differences between the means: $H_1: \mu_1^* \neq \mu_2^* \neq \dots \neq \mu_k^*$. The significance levels we aim for are at $\alpha_1 = 0.1$ and $\alpha_2 = 0.05$. Before each test we apply the Shapiro-Wilk test to establish the fact of data normality; it is the most adequate test for such an amount of data [22].

A good omnibus test, when the normality of the data cannot be guaranteed, is the Kruskal-Wallis test [23] which even in cases of near-normality can prove to be more sensitive than ANOVA. In the event of detecting significant differences between diffusion levels we will perform the Dunn's Pairwise Comparison method [24], the results of which should provide us with knowledge on the relations between different Diffusion

Depths d . We also must take into account the amount of data groups present in the test by applying the necessary Bonferroni correction. To achieve the agreed significance of $\alpha_1 = 0.1$ and $\alpha_2 = 0.05$ we must confirm results only at the significance $\alpha_{b1} = 0.01$ and $\alpha_{b2} = 0.005$ respectively, which is a very strong requirement.

All the experiments will be run on our proprietary software, written by us in the Java programming language. It was designed specifically for the simulation of the accelerated evolution of ACO algorithms in various settings. The machine used in testing is a PC Intel Pentium 4 630 at 3.00GHz with 4 GB of RAM on a 32bit Windows 7. Typically a single experiment might take up to 15 minutes of processing time.

A. In-width diffusion Experiment

With the configuration presented in Table III we attempted the recreation of the most basic, flood-like behavior in our system. The independent variable d is the TTL of Diffusion Ants, taking values $d | 0, d | 1, d | 2, d | 3$ and $d | 4$. Each test is repeated three times and the data will be presented as strained and non-strained, as explained in the section V.D.

The full dataset of the results is presented in Table VI. The normality was not achieved here due to a priori selection of the depth values; the Shapiro-Wilk test does not allow the assumption of normality with high certainty, see Table VII.

TABLE VI IN-WIDTH EXPERIMENT, FULL RESULTS

World Size	Diffusion Depth	Diffusion Size	HpH str.	HpH str.	HpH non-str.	HpH non-str.
-	d	-	μ_d	μ_d^*	$\bar{\mu}_d$	$\bar{\mu}_d^*$
1024	0	0	20.52	1.00	20.52	1.00
1024	1	4	17.74	0.87	17.58	0.86
1024	2	12	17.98	0.88	17.61	0.86
1024	3	24	19.07	0.93	17.30	0.84
1024	4	40	21.19	1.04	16.34	0.80
1600	0	0	21.78	1.00	21.78	1.00
1600	1	4	19.63	0.90	19.34	0.89
1600	2	12	18.81	0.86	17.89	0.82
1600	3	24	19.68	0.90	16.97	0.78
1600	4	40	26.64	1.22	18.22	0.84
2304	0	0	21.53	1.00	21.53	1.00
2304	1	4	20.38	0.95	19.96	0.93
2304	2	12	19.75	0.92	18.38	0.85
2304	3	24	22.53	1.05	18.32	0.85
2304	4	40	27.92	1.30	16.77	0.78
3136	0	0	21.56	1.00	21.56	1.00
3136	1	4	21.10	0.98	20.49	0.95
3136	2	12	19.37	0.90	17.59	0.82
3136	3	24	22.92	1.06	17.43	0.81
3136	4	40	33.05	1.53	17.31	0.80
4096	0	0	22.41	1.00	22.41	1.00
4096	1	4	21.30	0.95	20.51	0.92
4096	2	12	20.19	0.90	17.84	0.80
4096	3	24	25.95	1.16	18.38	0.82
4096	4	40	37.52	1.67	17.21	0.77

TABLE VII IN-WIDTH EXPERIMENT, NORMALITY TESTS

Shapiro-Wilk	Statistic	df	p-value
In-Width strain	0.749	25	0.000
In-Width non-strain	0.874	25	0.005

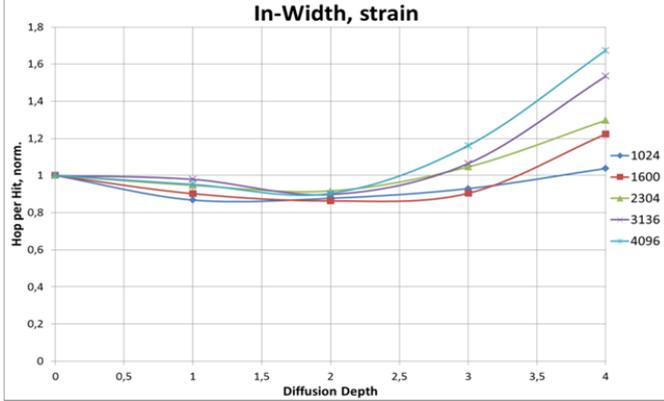


Fig. 3. In-Width Diffusion, strain graph

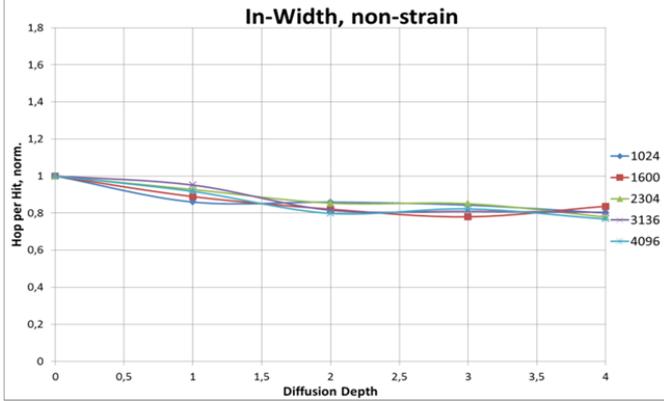


Fig. 4. In-Width Diffusion, non-strain graph

Kruskal-Wallis test has proven beyond any doubt that groups formed by different diffusion settings are significantly different and are disjointed populations of data points with respect to the HpH measure. The most interesting conclusions stem out of Table IX and Table X. For instance, in Table IX we see that the difference between non-str. $d | 2$ and $d | 0$ is of 12.100 ranks; from the corresponding field in the Table X we conclude that it is significant at $\alpha = 0.01$. In case of the non-strain values we confirm that even an algorithm as crude as k-flood is significantly better than the lack of diffusion for depths of diffusion equal or larger than $d | 2$, which are $d | 2$, $d | 3$ and $d | 4$. The strained data are less promising as there is no statistically significant benefit from the diffusion and therefore more elaborate algorithms are in place, if one cannot pinpoint system's downtimes to perform the diffusion and treat it as non-straining. We feel we need to point out however that the diffusion of size $d | 2$ was very nearly accepted as better than no diffusion, with $p - value = 0.036$. A simple look at the graphical representation of the strained results (Fig. 3) suggests that around the diffusion $d | 2$ there is a local minimum of net benefit at about 15% HpH. In case of the non-strained results (Fig. 4) no such minimum is observed and we assume that it is

located outside the examined scope; the best value is 23% lower than the diffusion-less system.

TABLE VIII IN-WIDTH EXPERIMENT, KRUSKAL-WALLIS TEST RESULTS

Kruskal-Wallis test	In-Width non-strain	In-Width strain
K (Observed value)	18.604	16.896
K (Critical value)	7.779	7.779
Df	4	4
p-value (Two-tailed)	0.001	0.002
Alpha	0.01	0.01

TABLE IX IN-WIDTH EXPERIMENT, DUNN'S COMPARISON, RANKS

		d 0	d 1	d 2	d 3	d 4
In-Width non-strain	d 0	0	4.600	12.100	13.800	17.500
	d 1	-4.600	0	7.500	9.200	12.900
	d 2	-12.100	-7.500	0	1.700	5.400
	d 3	-13.800	-9.200	-1.700	0	3.700
	d 4	-17.500	-12.900	-5.400	-3.700	0
In-Width strain	d 0	0	6.400	9.900	0.200	-6.900
	d 1	-6.400	0	3.500	-6.200	-13.300
	d 2	-9.900	-3.500	0	-9.700	-16.800
	d 3	-0.200	6.200	9.700	0	-7.100
	d 4	6.900	13.300	16.800	7.100	0

TABLE X IN-WIDTH EXPERIMENT, DUNN'S COMPARISON, P-VALUES

		d 0	d 1	d 2	d 3	d 4
In-Width non-strain	d 0	1	0.330	0.010^a	0.003^{ab}	0.000^{ab}
	d 1	0.330	1	0.092	0.039	0.004^{ab}
	d 2	0.010^a	0.092	1	0.702	0.225
	d 3	0.003^{ab}	0.039	0.702	1	0.406
	d 4	0.000^{ab}	0.004^{ab}	0.225	0.406	1
In-Width strain	d 0	1	0.175	0.036	0.966	0.144
	d 1	0.175	1	0.432	0.164	0.003^{ab}
	d 2	0.036	0.432	1	0.029	0.000^{ab}
	d 3	0.966	0.164	0.029	1	0.111
	d 4	0.144	0.003^{ab}	0.000^{ab}	0.111	1

Bonferroni corrected significance levels: ^a $\alpha_{p1} = 0.01$ ^b $\alpha_{p2} = 0.005$

B. In-depth diffusion Experiment

The in-depth diffusion experiment was executed according to the parameters established in Table IV. We manipulated the degree of diffusion by changing the TTL of Diffusion Ants over 5 values: $d | 0$, $d | 5$, $d | 10$, $d | 15$ and $d | 20$, it will be our independent variable d . Each test is repeated three times and the data will be presented as strained and non-strained, as it was explained in the section V.D

Similarly to the previous experiment the data cannot be considered normal, as it is shown in Table XII. We therefore must again run the statistical analysis with the Kruskal-Wallis test. With near perfect certainty (>99.99%, see Table XIII) it was confirmed that there are different populations of points within our results in terms of HpH measure. In case of In-Depth diffusion the strain put on the results is marginally low, from Table XI we conclude that even at the $d | 20$ Diffusion Depth the difference between strain and non-strain results is below 0.8 HpH. It comes as no surprise that strained and non-strained

results are very much alike, see Table XIV and Table XV for Dunn's pairwise comparison. We can conclude that the Diffusion Depths of $d \geq 15$ and higher are significantly better than no diffusion at all. It is also worth pointing out that the minute Diffusion Depth of $d \geq 10$ may be nearly considered appropriate as well. In these cases the improvement reaches quite an impressive value of 32% in the non-strained variant and up to 29% in the strained one. Both of these can be seen on Fig. 5 and Fig. 6 respectively. No clear local minimum was detected so pushing the Diffusion Depth to even higher values may be justified.

TABLE XI IN-DEPTH DIFFUSION, FULL RESULTS

World Size	Diffusion Depth	Diffusion Size	HpH str.	HpH str.	HpH non-str.	HpH non-str.
-	d	-	μ_d	μ_d^*	$\tilde{\mu}_d$	$\tilde{\mu}_d^*$
1024	0	0	19.46	1.00	19.46	1.00
1024	5	5	18.02	0.93	17.85	0.92
1024	10	10	15.89	0.82	15.59	0.80
1024	15	15	14.57	0.75	14.16	0.73
1024	20	20	14.04	0.72	13.52	0.70
1600	0	0	22.05	1.00	22.05	1.00
1600	5	5	19.42	0.88	19.13	0.87
1600	10	10	17.21	0.78	16.71	0.75
1600	15	15	15.82	0.72	15.14	0.69
1600	20	20	15.55	0.71	14.68	0.67
2304	0	0	21.99	1.00	21.99	1.00
2304	5	5	18.38	0.84	17.99	0.82
2304	10	10	18.91	0.86	16.69	0.76
2304	15	15	18.12	0.82	16.33	0.75
2304	20	20	16.50	0.75	15.20	0.69
3136	0	0	22.02	1.00	22.02	1.00
3136	5	5	20.18	0.92	19.6	0.89
3136	10	10	19.12	0.87	18.05	0.82
3136	15	15	18.29	0.83	16.80	0.76
3136	20	20	17.99	0.82	16.13	0.73
4096	0	0	21.91	1.00	21.91	1.00
4096	5	5	19.45	0.89	18.73	0.86
4096	10	10	18.21	0.83	16.90	0.78
4096	15	15	16.92	0.77	15.17	0.70
4096	20	20	18.27	0.84	15.86	0.73

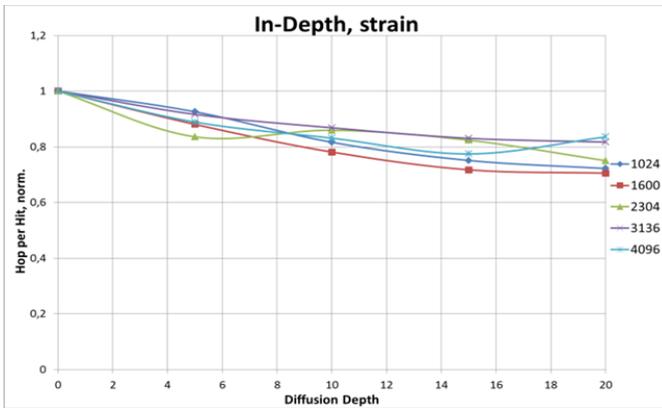


Fig. 5. In-Depth Diffusion, strain graph

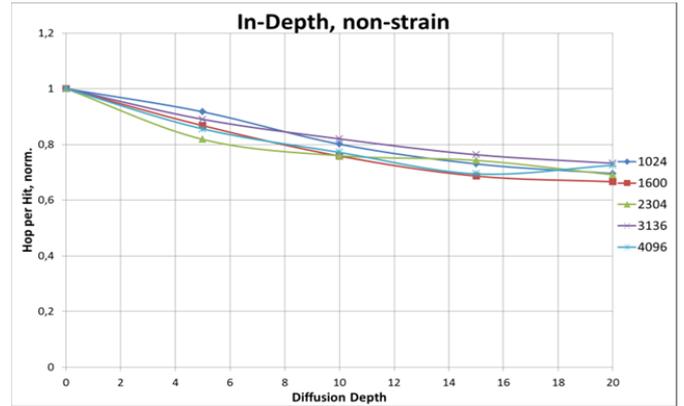


Fig. 6. In-Depth Diffusion, non-strain graph

TABLE XII IN-DEPTH EXPERIMENT, NORMALITY TESTS

Shapiro-Wilk	Statistic	df	p-value
In-Depth strain	0.926	25	0.069
In-Depth non-strain	0.885	25	0.009

TABLE XIII IN-DEPTH EXPERIMENT, KRUSKAL-WALLIS TEST RESULTS

Kruskal-Wallis test	In-Depth non-strain	In-Depth strain
K (Observed value)	20.677	18.166
K (Critical value)	7.779	7.779
Df	4	4
p-value (Two-tailed)	0.000	0.001
Alpha	0.01	0.01

TABLE XIV IN-DEPTH EXPERIMENT, DUNN'S COMPARISON, RANKS

		d 0	d 5	d 10	d 15	d 20
In-Depth non-strain	d 0	0	4.600	9.800	15.500	18.100
	d 5	-4.600	0	5.200	10.900	13.500
	d 10	-9.800	-5.200	0	5.700	8.300
	d 15	-15.500	-10.900	-5.700	0	2.600
	d 20	-18.100	-13.500	-8.300	-2.600	0
In-Depth strain	d 0	0	5.000	10.800	15.800	16.400
	d 5	-5.000	0	5.800	10.800	11.400
	d 10	-10.800	-5.800	0	5.000	5.600
	d 15	-15.800	-10.800	-5.000	0	0.600
	d 20	-16.400	-11.400	-5.600	-0.600	0

TABLE XV IN-DEPTH EXPERIMENT, DUNN'S COMPARISON, p-values

		d 0	d 5	d 10	d 15	d 20
In-Depth non-strain	d 0	1	0.330	0.038	0.001^{a,b}	0.000^{a,b}
	d 5	0.330	1	0.243	0.014	0.002^{a,b}
	d 10	0.038	0.243	1	0.201	0.062
	d 15	0.001^{a,b}	0.014	0.201	1	0.559
	d 20	0.000^{a,b}	0.002^{a,b}	0.062	0.559	1
In-Depth strain	d 0	1	0.290	0.022	0.001^{a,b}	0.001^{a,b}
	d 5	0.290	1	0.193	0.015	0.010^a
	d 10	0.022	0.193	1	0.262	0.209
	d 15	0.001^{a,b}	0.015	0.262	1	0.893
	d 20	0.001^{a,b}	0.010^a	0.209	0.893	1

Bonferroni corrected significance levels: ^a $\alpha_{b1} = 0.01$ ^b $\alpha_{b2} = 0.005$

VII. CONCLUSIONS / FUTURE WORK

In this work we have shown clearly that the introduction of our idea of the Diffusion Model Framework, with the objective of improving the re-convergence speed in a p2p environment managed by ACO algorithm, is beneficial. Even the crudest variant provided a certain net improvement to the system, in terms of HpH, and was helpful in combating the slow re-convergence.

The in-width version has proven to be a feasible solution in systems that have a downtime period. As mentioned before, those including day/night cycles, weekday/weekend cycles, etc. In this situation one might disregard the strain that the Diffusion Ant has on the system and successfully apply the in-width diffusion depth $d | 3$ or $d | 4$. To some extent the in-width diffusion depth $d | 2$ is nearly applicable to the strained system as well.

In the case of our in-depth strategy the results were much stronger. Not only the overall strain on the system was minimal, but also the improvement reached values as high as 30%. Regardless of whether we used strained or non-strained results we come to the conclusion that in-depth diffusion depths $d | 15$ and $d | 20$ are significantly better than no diffusion at all. This permits us to say that any system, that follows our prerequisites even loosely, would benefit from such an extension.

This paper focused on some basic ideas of the applicability of the diffusion. We want to take the subject further and test it in a much more general environment, where resources appear, disappear and modify. Our next step will also consist of applying the concept of diffusion to real-life networks, with where the dynamism is not limited to the nodes' content, but also to the nodes themselves which attach and detach from the system, creating breaks in the pheromone trail's continuity. We also intent to work on more elaborate and sophisticated diffusion algorithms, which revolve around the concept of the directed diffusion achieved through exploiting the topology of the network. This will lead to the creation of a lightweight diffusion algorithm that complements efficiently the classical ACO paradigm in p2p environments.

VIII. ACKNOWLEDGEMENTS

Kamil Krynicki is supported by a FPI fellowship from the Universitat Politècnica de Valencia with reference number 3117. This work received financial support from the Spanish Ministry of Education under the National Strategic Program of Research and Project TSI2010-20488.

IX. REFERENCES

- [1] M. Dorigo, V. Maniezzo, and A. Colomi, The Ant System: Optimization by a Colony of Cooperating Agents, *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, Vol. 26, No. 1, pp. 1-13, 1996.
- [2] E. Michlmayr, S. Graf, W. Siberski, and W. Nejdil, Query Routing with Ants, in *Proceedings of the 1st Workshop on Ontologies in P2P Communities, ESWC2005*, Heraklion, Greece, 2005.
- [3] G. Dong, WW. Guo, and K. Tickle, Solving the traveling salesman problem using cooperative genetic ant systems, *Expert Systems with Applications*, Vol. 39, No. 5, pp. 5006-5011, April 2012.
- [4] A. O. Bozdogan and M. Efe, Improved assignment with ant colony optimization for multi-target tracking, *Expert Systems with Applications*, Vol. 38, No. 8, pp. 9172-9178, August 2011.
- [5] J. A. Mocholi, J. J. Martínez, A. Catalá, and E. Navarro, An emotionally biased ant colony algorithm for pathfinding in games, *Expert System With Applications*, Vol. 37, No. 7, pp. 4921-4927, July 2010.
- [6] J. A. Mocholi, J. J. Martínez, K. Krynicki, A. Catalá, A. Picón, and A. Cadenas, Learning semantically-annotated routes for context-aware recommendations on map navigation system, *Applied Soft Computing*, Vol. 12, No. 9, pp. 3088-3098, September 2012.
- [7] J. J. Martínez, J. A. Mocholi, A. Catalá, and E. Navarro, Digital ants as the best cicerones for museum visitors, *Applied Soft Computing*, Vol. 11, No. 1, pp. 111-119, January 2011.
- [8] K. Krynicki, J. Jaen, and J. A. Mocholi, On the performance of ACO-based methods in P2P resource discovery, under review, *Applied Soft Computing*, momo02.dsic.upv.es/download/Static_ant.pdf.
- [9] M. Dorigo and G. Di Caro, The Ant Colony Optimization Meta-Heuristic, in *New Ideas in Optimization*.: McGraw-Hill, 1999, pp. 11-32.
- [10] E. Michlmayr, Ant Algorithms for Self-Organization in Social Networks. Vienna, Austria: Vienna University of Technology, 2006, Ph.D.
- [11] K. Krynicki, J. Jaen, and J. A. Mocholi, On the performance of ACO for resource searching in dynamic peer-to-peer grids, under review, *IEEE Trans. Evol. Comput.*, momo02.dsic.upv.es/download/Dynamic_ant.pdf.
- [12] A. Crespo and H. Garcia-Molina, "Semantic Overlay Networks for P2P Systems", Computer Science Department, Stanford University, Stanford, Technical report 2002.
- [13] S. Hautphenne, K. Leibnitz, and MA. Remiche, Extinction probability in peer-to-peer file diffusion, *ACM SIGMETRICS Performance Evaluation Review*, Vol. 34, No. 2, pp. 3-4, September 2006.
- [14] T. Hofffeld, K. Leibnitz, R. Pries, K. Tutschku, P. Tran-Gia, and K. Pawlikowski, Information Diffusion in eDonkey Filesharing Networks, in *Proceedings of the ATNAC 2004*, Sydney, 2004, p. 8.
- [15] D. Shi, J. Yin, J. Qian, Y. Li, and J. Dong, JTangPS: Content-Based Publish/Subscribe over Structured P2P Networks, *Advances in Intelligent and Distributed Computing Studies in Computational Intelligence*, Vol. 78, pp. 187-196, 2008.
- [16] A. Colomi, M. Dorigo, and V. Maniezzo, Distributed Optimization by Ant Colonies, in *Première Conférence Européenne Sur la Vie Artificielle*, Paris, France, 1991, pp. 134-142.
- [17] KM. Salama, AM. Abdelbar, FEB. Otero, and AA. Freitas, Utilizing multiple pheromones in an ant-based algorithm for continuous-attribute classification rule discovery, *Applied Soft Computing*, Vol. 13, No. 1, pp. 667-675, January 2013.
- [18] An Efficient Scheme for Query Processing on Peer-to-Peer Networks. M. T. Prinkey. (2001) <http://aeolusres.homestead.com/files/index.html>
- [19] Association for Computing Machinery. (1998) ACM Computing Classification System (ACM CSS).
- [20] S. Zhao, D. Stutzbach, and R. Rejaie, Characterizing Files in the Modern Gnutella Network: A Measurement Study, in *Proceedings of SPIE/ACM Multimedia Computing and Networking*, 2006.
- [21] C. Gómez Santillán, L. Cruz Reyes, E. Meza Conde, E. Schaeffer, and G. Castilla Valdez, A Self-Adaptive Ant Colony System for Semantic Query Routing Problem in P2P Networks, *Computación y Sistemas*, Vol. 13, No. 4, pp. 433 - 448, 2010.
- [22] S. S. Shapiro and M. B. Wilk, An analysis of variance test for normality (complete samples), *Biometrika*, Vol. 52, No. 3-4, pp. 591-611, 1965.
- [23] W. H. Kruskal and W. A. Wallis, Use of Ranks in One-Criterion Variance Analysis, *Journal of the American Statistical Association*, Vol. 47, No. 260, pp. 583-621, December 1952.
- [24] O.J. Dunn, Multiple Comparisons Among Means, *Journal of the American Statistical Association*, Vol. 56, No. 293, pp. 52-64, 1961.